Intermediate Statistical Methods
Lecture Notes for CSC/DSC 265/465
SPRING 2017

Anthony Almudevar
Department of Biostatistics and Computational Biology,
University of Rochester
Rochester, NY 14642, USA

February 2, 2017

Contents

Part I

Statistical Methodology

Chapter 1

ANOVA

We often have situations in which we have k random samples from k distinct populations with population means $\mu_1, \ldots, \mu_k$. Interest is then in testing the hypothesis

$$H_o : \mu_1 = \mu_2 = \ldots = \mu_k$$
$$H_a : \mu_i \neq \mu_j \text{ for some } i,j$$

In other words, are there some differences between the means (H_a) or are they all the same (H_o).

1.1 Methodology

The technique we use is referred to as *analysis of variance*, or *ANOVA*. The data then has the following structure

Pop'n	Pop'n Mean	Sample Size	Sample	Sample Mean	Sum of Squares
1	μ_1	n_1	$y_{11}, y_{12}, \ldots, y_{1n_1}$	$\bar{y}_1$	$\sum_{i=1}^{n_1}(y_{1i} - \bar{y}_1)^2$
2	μ_2	n_2	$y_{21}, y_{22}, \ldots, y_{2n_2}$	$\bar{y}_2$	$\sum_{i=1}^{n_2}(y_{2i} - \bar{y}_2)^2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
k	μ_k	n_k	$y_{k1}, y_{k2}, \ldots, y_{kn_k}$	$\bar{y}_k$	$\sum_{i=1}^{n_k}(y_{ki} - \bar{y}_k)^2$

The groups may be referred to as *treatments*. Sometimes it is convenient to refer to a treatment as a *factor* or *factor variable*. The observations y_{ij} are then *responses*, and μ_i is a *mean response*. Here, we only have one factor, so the procedure is referred to as *one-way ANOVA*. If the sample sizes n_i are equal, we may refer to a *balanced design*.

We also have the *total mean*

$$\begin{aligned} \bar{y} &= \frac{\text{sum of all observations}}{n_1 + n_2 + \ldots + n_k} \\ &= \frac{n_1\bar{y}_1 + n_2\bar{y}_2 + \ldots + n_k\bar{y}_k}{n_1 + n_2 + \ldots + n_k} \end{aligned}$$

In order to develop a test statistic for the hypothesis, we define the *treatment sum of squares*

$$\text{SST} = \sum_{i=1}^{k} n_i(\bar{y}_i - \bar{y})^2$$

and the *error sum of squares*

$$\text{SSE} = \sum_{i=1}^{k}\sum_{j=1}^{n_i}(y_{ij} - \bar{y}_i)^2.$$

It can be shown that if define the *total sum of squares* to be

$$\text{SSTO} = \sum_{i=1}^{k}\sum_{j=1}^{n_i}(y_{ij} - \bar{y})^2.$$

then

$$\text{SSTO} = \text{SST} + \text{SSE}.$$

The test statistic we use is then

$$F_{obs} = \frac{SST/(k-1)}{SSE/(n-k)}$$

where

$$n = n_1 + n_2 + \ldots + n_k.$$

Given the form of SST we can see that if there are large differences among the sample means, F_{obs} will tend to be larger. To reject the null hypothesis we use the observed significance level defined by

$$\alpha_{obs} = P(F_{k-1,n-k} > F_{obs})$$

where F_{ν_1,ν_2} is a random variable with an *F distribution* with ν_1 *numerator degrees of freedom* and ν_2 *denominator degrees of freedom*. Most statistical software packages will calculate this significance level.

Example 1.1. This example is due to Johnson and Bhattacharya (*Statistics: Principles and Methods, Wiley, 3rd edition*).

In an effort to improve the quality of recording tapes, the effects of four kinds of coatings A, B, C and D on reproduction quality are assessed by applying each to a separate sample of tape and measuring the resulting distortion. The results are given in the following table.

Coating	Sample	Sample Mean	Sum of Squares
A	10, 15, 8, 12, 15	$\bar{y}_1 = 12$	$\sum_{i=1}^{5}(y_{1i} - \bar{y}_1)^2 = 38$
B	14, 18, 21, 15	$\bar{y}_2 = 17$	$\sum_{i=1}^{4}(y_{2i} - \bar{y}_2)^2 = 30$
C	17, 16, 14, 15, 17, 15, 18	$\bar{y}_3 = 16$	$\sum_{i=1}^{7}(y_{3i} - \bar{y}_3)^2 = 12$
D	12, 15, 17, 15, 16, 15	$\bar{y}_4 = 15$	$\sum_{i=1}^{6}(y_{4i} - \bar{y}_4)^2 = 14$

We therefore have

$$\begin{aligned} k &= 4 \\ n &= n_1 + n_2 + n_3 + n_4 \\ &= 5 + 4 + 7 + 6 \\ &= 22 \\ \hat{y} &= \frac{n_1\bar{y}_1 + n_2\bar{y}_2 + n_2\bar{y}_2 + n_4\bar{y}_4}{n_1 + n_2 + \ldots + n_k} \\ &= \frac{5 \times 12 + 4 \times 17 + 7 \times 16 + 6 \times 15}{22} \\ &= 15. \end{aligned}$$

The sums of squares are given by

$$\begin{aligned} \text{SSE} &= \sum_{i=1}^{5}(y_{1i} - \bar{y}_1)^2 + \sum_{i=1}^{4}(y_{2i} - \bar{y}_2)^2 + \sum_{i=1}^{7}(y_{3i} - \bar{y}_3)^2 + \sum_{i=1}^{6}(y_{4i} - \bar{y}_4)^2 \\ &= 38 + 30 + 12 + 14 \\ &= 94 \end{aligned}$$

and

$$\begin{aligned} \text{SST} &= n_1(\bar{y}_1 - \bar{y})^2 + n_2(\bar{y}_2 - \bar{y})^2 + n_3(\bar{y}_3 - \bar{y})^2 + n_4(\bar{y}_4 - \bar{y})^2 \\ &= 5(-3)^2 + 4(2)^2 + 7(1)^2 + 6(0)^2 \\ &= 68 \end{aligned}$$

giving test statistic

$$\begin{aligned} F_{obs} &= \frac{SST/(k-1)}{SSE/(n-k)} \\ &= \frac{68/3}{94/18} \\ &= 4.34 \end{aligned}$$

The observed significance level can be calculated using the appropriate table or with a computer program, and can be found to be

$$\alpha_{obs} = .018$$

meaning that there is evidence that the four means are not identical. □

1.2 ANOVA Table

The results of an ANOVA calculation are usually summarized in an *ANOVA summary table* of the following form

Source	SS	df	MS	
Between Treatment (or Treatment)	SST	$k-1$	$MST = \frac{SST}{k-1}$	$F = \frac{MST}{MSE}$
Within Treatment (or Error)	SSE	$n-k$	$MSE = \frac{SSE}{n-k}$	
Total	SSTO	$n-1$		

Where

k	Number of groups	
n_i	Sample size of group i	
n	Total sample size	$n_1 + \ldots + n_k$
$\bar{y}_i$	Sample mean of group i	
$\bar{y}$	Total sample mean	$\frac{n_1\bar{y}_1+\ldots+n_k\bar{y}_k}{n_1+\ldots+n_k}$
SSE	Error sum of squares or Within Treatment SS	$\sum_{i=1}^{k}(n_k-1)s_k^2$
SST	Treatment sum of Squares or Between Treatment SS	$\sum_{i=1}^{k} n_k(\bar{y}_k-\bar{y})^2$
SSTO	Total sum of squares	SST + SSE
MSE	Mean error sum of squares or Mean within Treatment SS	$\frac{SSE}{n-k}$
MST	Mean treatment sum of squares or Mean between Treatment SS	$\frac{SST}{k-1}$
F	F-ratio	$\frac{MST}{MSE}$

For the tape coating problem

$$\begin{aligned} n &= 22 \\ k &= 4 \\ SSE &= 94 \\ MSE &= 94/(n-k) \\ &= 5.22 \\ SST &= 68 \\ MST &= 68/(k-1) \\ &= 22.67 \\ F &= 4.34 \end{aligned}$$

The ANOVA table for this problem is then

Source	SS	df	MS	
Between Treatment	68	3	22.67	4.34
Within Treatment	94	18	5.22	
Total	162	21		

Recall that is it assumed that each sample comes from a population with possibly differing means, but with one common variance σ^2. It can be shown that MSE is an estimator of σ^2. In fact, if there are $k = 2$ groups then the MSE is identical to the pooled sample variance S_p^2 and plays the same role when $k > 2$.

1.3 Bonferroni Correction for Multiple Comparisons

We often encounter a situation in which we wish to report several confidence intervals or hypothesis tests. If we use a confidence level $(1 - \alpha)$ for each confidence interval, or a significance level of α for each hypothesis test, we must consider the fact that the probability of at least one error among all inference statements will be greater than α.

A number of procedures exist with which to control *familywise error rate* (FWE), that is, the probability that among a set of m inference statements there is at least one error (the term *group* is sometimes used in place of 'familywise'). The commonly used convention is that an error rate suitable for a single inference procedure should also be applied to multiple inferences, so that the FWE is commonly set to $\alpha_{FWE} = 0.05$. We can also refer to familywise (or group) confidence level $1 - \alpha_{FWE}$.

A large number of *multiple comparison* procedures exist, some specialized and others general. Probably the most commonly encountered method is known as the *Bonferroni correction procedure* (BCP), which is applicable, in principle, to any multiple comparison model. Recall Booles's inequality:

$$P\left(\cup_{i=1}^{m} E_i\right) \leq \sum_{i=1}^{m} P\left(E_i\right).$$

Suppose we are given m level $(1 - \alpha)$ confidence intervals

$$E_i = \{ \text{ the } i\text{th CI is incorrect } \}.$$

Then $P(E_i) = \alpha$ and

$$P\left(\cup_{i=1}^{m} E_i\right) \leq m\alpha. \tag{1.1}$$

This means that all m confidence intervals are correct with a probability of at least $1 - m\alpha$. Therefore, in order to achieve a FWE of α_{FWE}, we would need to use a confidence level of $(1 - \alpha_{FWE}/m)$.

Example 1.2. Normally, to achieve a confidence level of 95% for a confidence interval

$$\bar{X} \pm z_{\alpha/2}\, \sigma/\sqrt{n} \tag{1.2}$$

we would set $\alpha = 0.05$ and therefore use critical value $z_{0.025} \approx 1.96$. If we wanted to simultaneously report $m = 4$ confidence intervals with FWE $\alpha_{FWE} = 0.05$, we would build separate level $(1-\alpha/m)$ confidence intervals. From (1.1), this would give

$$\alpha_{FWE} \leq m\alpha/m = \alpha,$$

so that it would be appropriate to set $\alpha = \alpha_{FWE}$. Therefore, in (1.2) we would use the critical value

$$z_{\alpha_{FWE}/(2m)} = z_{.05/8} = z_{.00625} \approx 2.5$$

for $\alpha_{FWE} = 0.05$. However, construction of the confidence intervals uses the same methodology once the Bonferroni correction has been applied.

The sample principle applies to hypothesis tests. If we want to report m hypothesis tests with a familywise Type I error of α_{FWE} (that is, at least one Type I error among the m tests), then each test must be carried out with a significance level of α_{FWE}/m. □

1.4 *Post hoc* Analysis in ANOVA

If we conclude that there is some difference between means using the F-test, then we may wish to further explore how the means differ. A common way to achieve this is through the use of *pairwise multiple comparisons.*

Using the BCP we have

$$\bar{y}_i - \bar{y}_j \pm t_{\alpha/(m2),n-k}\sqrt{MSE\left(\frac{1}{n_i}+\frac{1}{n_j}\right)}$$

where m is the number of comparisons we wish to make (we need not be interested in all available comparisons).

If μ_i and μ_j are two group means, then a confidence interval for $\mu_i - \mu_j$ is given by

$$\bar{y}_i - \bar{y}_j \pm q_\alpha\sqrt{\frac{MSE}{2}\left(\frac{1}{n_i}+\frac{1}{n_j}\right)}$$

where q_α is the critical value from the *studentized range* distribution with $k-1$ treatment degrees of freedom and $n-k$ error degrees of freedom (these have the same interpretation as the numerator and denominator degrees of freedom for the F-distribution). Tables for these critical values are available in most textbooks. This is known as *Tukey's pairwise procedure* or sometimes the *Tukey-Kramer pairwise procedure*. It should be noted that the procedure is approximate, unless the design is balanced.

Note that there will be $k(k-1)/2$ comparisons. It needs to be stressed that the confidence level $1-\alpha$ represents the probability that *all* $k(k-1)/2$ confidence intervals are correct, and not just each one taken individually.

To continue with the tape coating problem, we have 18 error degrees of freedom and 3 treatment degrees of freedom so if we want a 95% confidence interval for all pairwise comparisons simultaneously we set

$$q_{0.05} = 4.00$$

and we also have

$$MSE = 5.22$$

with

Coating	n_i	Sample mean
A	$n_1 = 5$	$\bar{y}_1 = 12$
B	$n_2 = 4$	$\bar{y}_2 = 17$
C	$n_3 = 7$	$\bar{y}_3 = 16$
D	$n_4 = 6$	$\bar{y}_4 = 15$

giving pairwise confidence intervals

Pair	Confidence Interval
$\mu_1 - \mu_2$:	-5 ± 4.33
$\mu_1 - \mu_3$:	-4 ± 3.78
$\mu_1 - \mu_4$:	-3 ± 3.91
$\mu_2 - \mu_3$:	1 ± 4.06
$\mu_2 - \mu_4$:	2 ± 4.17
$\mu_3 - \mu_4$:	1 ± 3.59

We may conclude that μ_1 is significantly different from μ_2 and μ_3 but can make no other conclusions based on this procedure.

Note that there are many pairwise procedures, most notably the *Scheffe test* which is very conservative. This provides a FWE of α_{FWE} of confidence intervals for all *contrasts*

$$C = \sum_{i=1}^{n} c_i \mu_i, \text{ where } \sum_{i=1}^{k} c_i = 0.$$

A pairwise comparison of the form $\mu_i - \mu_j$ is a contrast of the form $c_i = 1$, $c_j = -1$, $c_k = 0$ for $k \neq i, j$.

1.5 Nonparametric ANOVA

Recall that ANOVA may be thought of as an extension of the two-sample t-test for differences in mean to a K-sample test for differences in means, under the assumptions that variances are equal. Similarly, the *Kruskal-Wallis test* is an extension of the Wilcoxon rank sum test to K samples, and may be considered a nonparametric alternative to ANOVA. Under the null hypothesis, K samples are taken from K identical distributions (not necessarily normally distributed). We won't discuss the details of this test, but will note that the Kruskal-Wallis test is implemented in most statistical software packages. It would be appropriate to use whenever ANOVA might be used, but does not assume that the data is normally distributed.

1.6 Assumptions

The essential assumptions made for ANOVA are that population i has a $N(\mu_i, \sigma^2)$ distribution. The means may differ between populations but variances do not. In addition, each sample is a true random sample, and the samples are independent of each other.

Of course, ANOVA is a technique which has received a great deal of attention by statistical practitioners, so that there are a wide variety of techniques which may be used when these assumptions do not hold.

1.7 ANOVA in R

To fit an ANOVA model in R, we express the data as such a model. The variable Y is a single vector which contains all the variable. X is a single vector of *factors* which define the treatments. For example, to set up an ANOVA model in R, we can use the commands:

```
> y1 = rnorm(5,mean=10,sd=2.4)
> y2 = rnorm(6,mean=20,sd=2.4)
> y3 = rnorm(4,mean=20,sd=2.4)
>
> y = c(y1, y2, y3)
> x = c(rep(1,5), rep(2,6), rep(3, 4))
> x = as.factor(x)
> cbind(x,y)
      x         y
 [1,] 1  6.526123
 [2,] 1 10.639951
 [3,] 1  8.128591
 [4,] 1 10.922928
 [5,] 1 13.934701
 [6,] 2 20.502000
 [7,] 2 22.109955
 [8,] 2 22.575551
 [9,] 2 23.177065
[10,] 2 19.509436
[11,] 2 19.890914
[12,] 3 15.414790
[13,] 3 18.572681
[14,] 3 20.668094
[15,] 3 15.777068
>
```

This simulates an ANOVA model with $k = 3$ treatments, identified in the factor variable x. The sample sizes are $n_1 = 5$, $n_2 = 6$, $n_3 = 4$, with means $\mu_1 = 10$, $\mu_2 = \mu_3 = 20$. The common variance is $\sigma^2 = 2.4^2$.

It is usually a good idea to plot the data, and this can be done using boxplots. The R model notation can be used to separate the groups:

```
> boxplot(y ~ x)
```

The plot is shown in Figure 1.1.

There are several ways to fit an ANOVA model in R. One dedicated function is `aov()` used as follows:

```
> fit = aov(y ~ x)
> summary(fit)
            Df Sum Sq Mean Sq F value   Pr(>F)
```

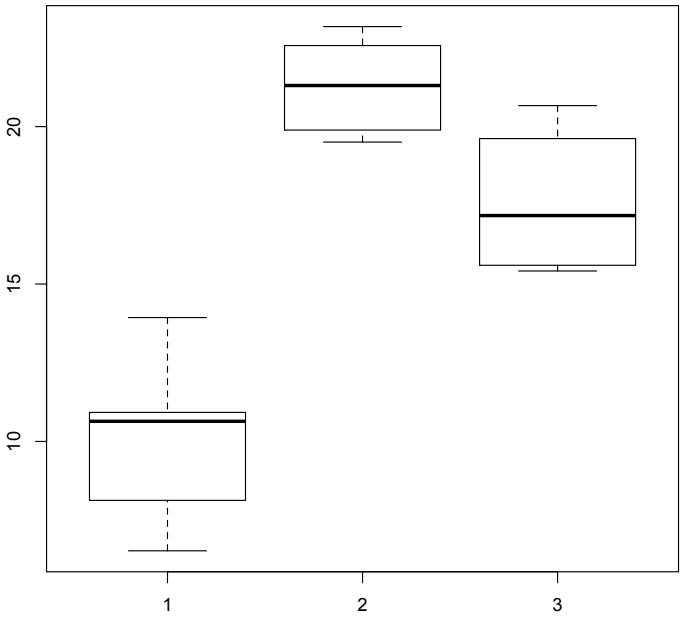

Figure 1.1: Multiple boxplots for ANOVA example

```
x            2  352.0   176.0   33.85 1.17e-05 ***
Residuals   12   62.4     5.2
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Note that the fit itself can be stored as an object, which is generally good practice. Fit objects can then be used as input for generic functions, which provide summaries for the appropriate type of object. For example `summary()` gives for an `aov()` object the standard ANOVA table.

Tukey's pairwise procedure is also available for an ANOVA fit using the `TukeyHSD()` function (HSD refers to 'Honest Significant Difference):

```
> fit.Tukey = TukeyHSD(fit)
> fit.Tukey
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = y ~ x)

$x
         diff       lwr        upr     p adj
2-1 11.263695  7.579838 14.9475516 0.0000085
```

```
3-1  7.577699  3.496636 11.6587623 0.0009009
3-2 -3.685995 -7.613000  0.2410094 0.0665423

>
```

Notice that a new `R` object was produced by the `TukeyHSD()` function. If we use the generic `plot()` function we get the following plot (Figure 1.2):

```
> plot(fit.Tukey)
```

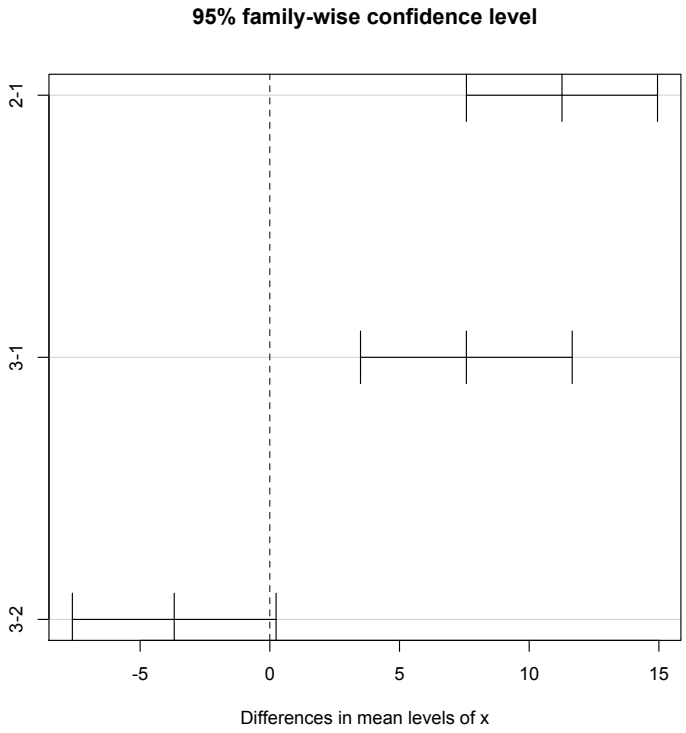

Figure 1.2: Graphical representation of Tukey's pairwise procedure.

1.8 Equality of Variances

We have seen how to test for the equality of two variances. *Bartlett's Test* is a generalization to k variances suitable for ANOVA. This is available using the `bartlett.test()`, using the same model notation. This may use the same model notation:

```
  > bartlett.test(y ~ x)

Bartlett test of homogeneity of variances
```

```
data:  y by x
Bartlett's K-squared = 1.5712, df = 2, p-value = 0.4558
```

The large p-value means that the hypothesis of equality or variances (also known as *homoscedasticity*, as opposed to *heteroscedasticity*) was not rejected.

It is also possible to use `bartlett.test()` by input a list of samples:

```
> y.list = list(y1, y2, y3)
> y.list
[[1]]
[1]  6.526123 10.639951  8.128591 10.922928 13.934701

[[2]]
[1] 20.50200 22.10996 22.57555 23.17706 19.50944 19.89091

[[3]]
[1] 15.41479 18.57268 20.66809 15.77707

> bartlett.test(y.list)

Bartlett test of homogeneity of variances

data:  y.list
Bartlett's K-squared = 1.5712, df = 2, p-value = 0.4558

>
```

However, `aov()` cannot be used this way.

We finally note that a model can be converted to a list using the `split()` function:

```
> split(y,x)
$`1`
[1]  6.526123 10.639951  8.128591 10.922928 13.934701

$`2`
[1] 20.50200 22.10996 22.57555 23.17706 19.50944 19.89091

$`3`
[1] 15.41479 18.57268 20.66809 15.77707

>
```

1.9 The Kruskal-Wallis Test for Nonparametric ANOVA

We have briefly introduced the Kruskal-Wallis test as an extention of the rank sum procedure to more than 2 samples. This is implemented in `R` using the `kruskal.test()` function, which is similar to `aov()`. For example, consider the simulated data:

```
> y1 = rnorm(5,mean=10,sd=2.4)
> y2 = rnorm(6,mean=20,sd=2.4)
> y3 = rnorm(4,mean=20,sd=2.4)
>
> y = c(y1, y2, y3)
> x = c(rep(1,5), rep(2,6), rep(3, 4))
> x = as.factor(x)
> cbind(x,y)
      x         y
 [1,] 1 11.695035
 [2,] 1  7.967687
 [3,] 1  8.891760
 [4,] 1 12.476237
 [5,] 1 10.996793
 [6,] 2 21.324256
 [7,] 2 19.594350
 [8,] 2 15.141688
 [9,] 2 21.956811
[10,] 2 19.251928
[11,] 2 16.064112
[12,] 3 21.472492
[13,] 3 20.310484
[14,] 3 26.817237
[15,] 3 20.906802
>
> boxplot(y ~ x)
>
```

The boxplot is shown in Figure 1.3. The data may be input into the `kruskal.test()` function as a model:

```
> fit = kruskal.test(y ~ x)
> summary(fit)
          Length Class  Mode
statistic 1      -none- numeric
parameter 1      -none- numeric
p.value   1      -none- numeric
method    1      -none- character
data.name 1      -none- character
> fit

Kruskal-Wallis rank sum test

data:  y by x
Kruskal-Wallis chi-squared = 10.3958, df = 2, p-value = 0.005528
```

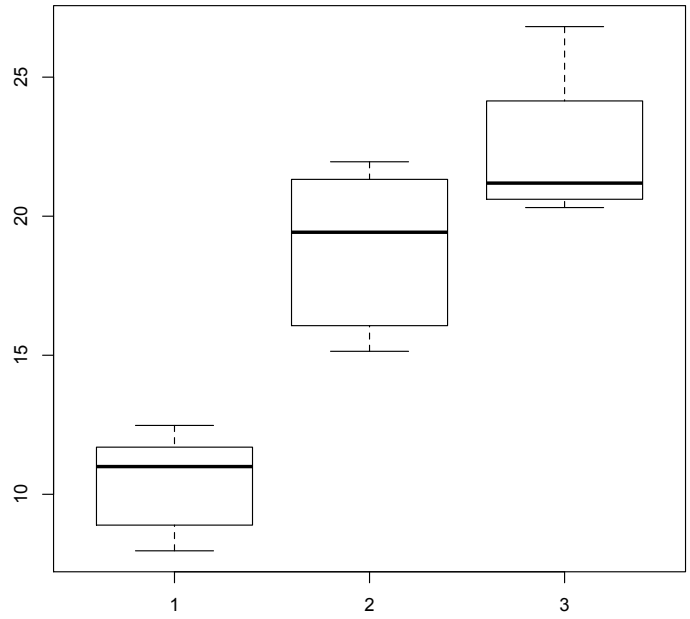

Figure 1.3: Multiple boxplots for Kruskal-Wallis test.

```
> stat = fit$statistic
> df = fit$parameter
> 1-pchisq(stat,df)
Kruskal-Wallis chi-squared
               0.005528069
>
```

However, in this case the `summary()` function only lists the labels which define the list elements of the fit object. These labels provide access to the output quantities. For example, the significance level is calculated from a χ^2 statistic with 2 degrees of freedom. We can access the statistic and the degrees of freedom by references to `fit$statistic` and `fit$parameter`. We have justy illustrated this by recalculating the p-value from the output.

The function `kruskal.test()` also accepts multiple samples in list form. In addition, it has a `g` option which defines groups, permitting the data to be entered as a single array"

```
> y.list = list(y1, y2, y3)
> y.list
[[1]]
[1] 11.695035  7.967687  8.891760 12.476237 10.996793

[[2]]
```

```
[1] 21.32426 19.59435 15.14169 21.95681 19.25193 16.06411

[[3]]
[1] 21.47249 20.31048 26.81724 20.90680

> kruskal.test(y.list)

Kruskal-Wallis rank sum test

data:  y.list
Kruskal-Wallis chi-squared = 10.3958, df = 2, p-value = 0.005528

> kruskal.test(y, g = x)

Kruskal-Wallis rank sum test

data:  y and x
Kruskal-Wallis chi-squared = 10.3958, df = 2, p-value = 0.005528

>
```

Chapter 2

Linear Regression - Introduction

Consider the scatter plot in Figure 2.1 representing 392 automobiles. The horizontal axis gives the engine displacement in cubic inches and the vertical axis gives horsepower.

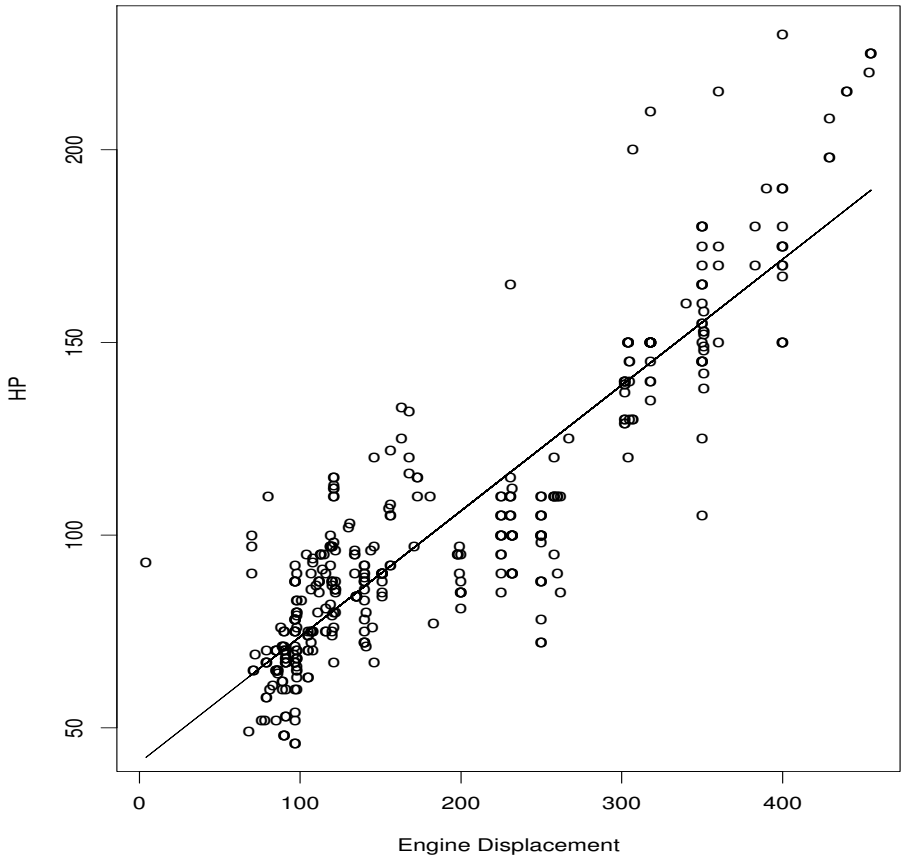

Figure 2.1: Scatter plot of automobile data

There seems to be a definite increasing trend in horsepower as engine displacement increases. If we set

$$\begin{aligned} X &= \text{Engine Displacement} \\ Y &= \text{Horsepower} \end{aligned}$$

then we should have approximately a linear relationship

$$Y = \beta_0 + \beta_1 X$$

where β_0 and β_1 are *coefficients* to be determined from the data. Looking at the scatter plot we see that the relationship will not be exact, so we introduce a random term ϵ (*epsilon*) into the equation.

$$Y = \beta_0 + \beta_1 X + \epsilon.$$

The usual terminology is to refer to Y as the *dependent variable* and to X as the *independent variable* or *predictor*. Here, there is only one predictor X, so the model is termed *simple linear regression*. When more predictors are used, for example $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$, the model is termed *multiple linear regression*.

If we have n pairs of dependent and independent observations

$$(X_1, Y_1), \ldots, (X_n, Y_n)$$

then the regression equation can be written in terms of the sample

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \;\; i = 1, \ldots, n. \tag{2.1}$$

Here, we assume that the *error terms* $\epsilon_1, \ldots, \epsilon_n$ form a random sample from $N(0, \sigma^2)$. We do not observe the error terms (unlike X_i and Y_i), but we can estimate σ^2.

The *linear least squares coefficients* are given by

$$\begin{aligned} \hat{\beta}_1 &= \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} \\ \hat{\beta}_0 &= \bar{Y} - \hat{\beta}_1 \bar{X}. \end{aligned}$$

With a large enough sample size we have estimates

$$\hat{\beta}_0 \approx \beta_0 \text{ and } \hat{\beta}_1 \approx \beta_1,$$

giving the estimated relationship between X and Y

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X.$$

We also have the *predicted responses*

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i.$$

for each sample pair $i = 1, \ldots, n$. Of course, we may construct a predicted response for a predictor value not represented in the sample, that is,

$$\hat{Y}_x = \hat{\beta}_0 + \hat{\beta}_1 x$$

is the predicted response for a predictor value $X = x$.

In the above example we have

$$\begin{aligned} \hat{\beta}_1 &= 0.327 \\ \hat{\beta}_0 &= 41.002, \end{aligned}$$

and this line is drawn in Figure 2.1.

Most statistical software implements linear regression, giving output in the following format

Model		Unstandardized Coefficients B	Std. Error	t	Sig.
1	(Constant)	41.002	1.792	22.884	.000
	Engine Displacement (cu. inches)	.327	.008	40.500	.000

Least squares coefficients can be taken directly from this table. If we wish to construct a level $(1-\alpha)100\%$ confidence interval for β_0 and β_1 we may use

$$\begin{aligned} \hat{\beta}_0 &\pm t_{n-2,\alpha/2} \times \text{Std. Error for } \hat{\beta}_0 \\ \hat{\beta}_1 &\pm t_{n-2,\alpha/2} \times \text{Std. Error for } \hat{\beta}_1 \end{aligned}$$

where the standard error may be taken from the table. Note that the appropriate degrees of freedom for the t-distribution critical values are $n-2$. If n is very large, we may use the standard normal critical value $z_{\alpha/2}$ instead. For the above example we have 95% confidence intervals

$$\begin{aligned} CI_{.95} &= 41.002 \pm 1.96 \times 1.792 \\ &= 41.002 \pm 3.5 \end{aligned}$$

for β_0 and

$$\begin{aligned} CI_{.95} &= 0.327 \pm 1.96 \times 0.008 \\ &= 0.327 \pm 0.0157 \end{aligned}$$

for β_1.

An important hypothesis test is

$$\begin{aligned} H_o &: \beta_1 = 0 \\ H_a &: \beta_1 \neq 0. \end{aligned}$$

This tells us whether or not there is any relationship between the dependent and independent variable. The observed significance level can be read directly from the table in the last column. Here, the observed significance level is given as 0, which means that there is strong evidence of a linear relationship between engine displacement and horsepower.

2.1 Residuals

The basic assumption used here is that the error terms $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ where

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

are equivalent to a random sample from a normal distribution with mean 0 and variance σ^2. Implicit in this formulation is the assumption that there is a linear relationship between X and Y.

Of course, the ϵ_i's cannot be directly observed, but they can be estimated by the *residuals*, given by

$$e_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i = Y_i - \hat{Y}_i$$

once the regression has been calculated. There are several ways to use the residuals to check the assumptions.

1. Draw a scatter plot of the points $(e_i, \hat{Y}_i)$ where $\hat{Y}_i$ is the *predicted value*

   $$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i.$$

   If the assumptions are satisfied there should be no pattern.

   (a) Check to see if the variation of the residuals appears to increase or decrease systematically. If so, this means that the variance of the error terms is not constant.

   (b) If large groups of residuals located next to each other appear to be all above or all below zero, then the assumption that the error terms are independent of each other may be incorrect. This is a frequent occurrence when the X_i's represent sequential points in time.

   (c) If the residuals appear to suggest some functional form, then the assumption of a linear relationship between X and Y may be incorrect.

2. To check for the assumption of normality of the error terms, construct a normal probability plot of the residuals. Departures from linearity indicate departures from normality of the error terms.

As a final remark linear regression, like ANOVA, is a widely used tool, and many techniques exist which may used when some of these assumptions are not valid.

Example 2.1. To continue with the automobile section we present a residual plot and a normal probability plot (Figure 2.2).

The residual plot shows a somewhat different behavior below and above 100 horsepower, which might be investigated. Other than that, no systematic departure from the assumption of no pattern is indicated.

The normal probability plot is approximately linear, except for the two extreme regions. This indicates that the normal distribution might not be accurate for very small tail probabilities, but otherwise should suffice as an approximation.

Example 2.2. As a second example, a scatter plot is presented with a linear regression fit in Figure 2.3. From the scatter plot, we can see that although there is a strong relationship between miles per gallon and horsepower, it is not a strictly linear one. Accordingly, the residual plot indicates a systematic functional form, suggesting that a linear fit is not the appropriate one.

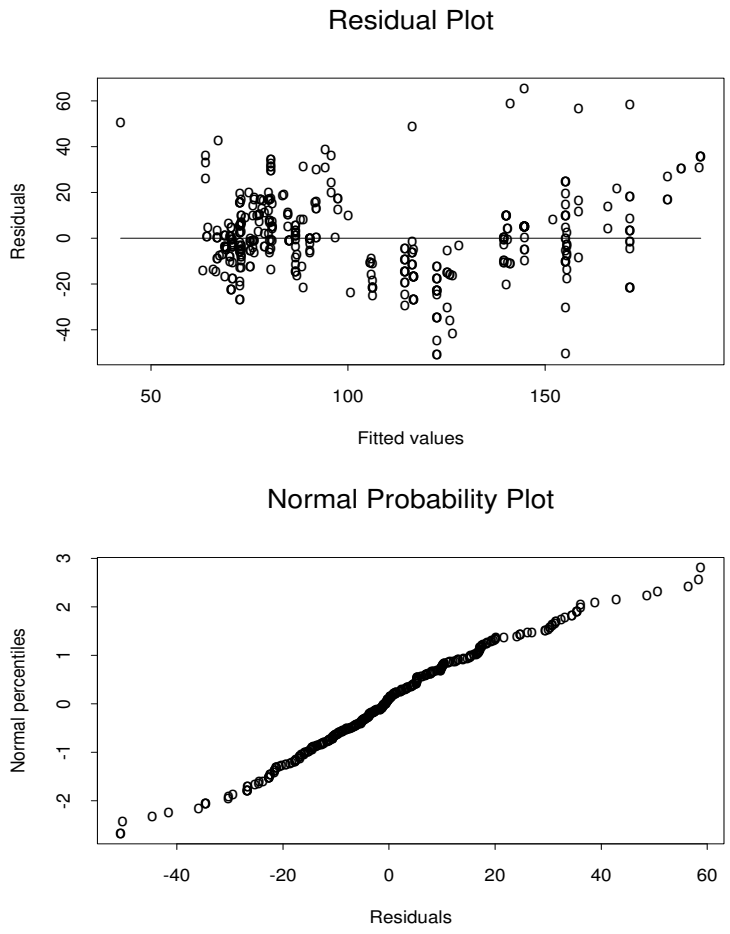

Figure 2.2: Residual plot and normal probability plot of residuals for Example 2.1

2.2 ANOVA approach

In linear regression, variation may be decomposed in a manner similar to that of ANOVA. We start with the *error sum of squares*

$$SSE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2,$$

and the *mean error sum of squares*

$$MSE = \frac{SSE}{n-2}.$$

The MSE is analogous to the MSE encountered in ANOVA, with $K = 2$ treatments corresponding to the 2 unknown parameters β_0 and β_1. In fact, the MSE functions as an estimate of the variance σ^2 encountered in the distribution $\epsilon_i \sim N(0, \sigma^2)$:

$$\hat{\sigma}^2 = MSE \approx \sigma^2.$$

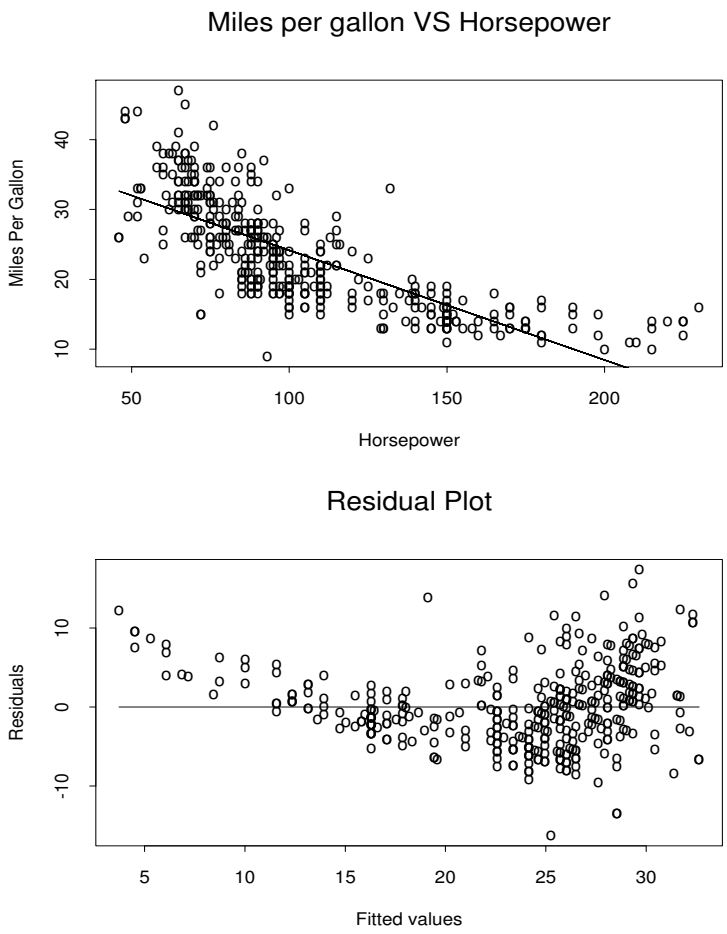

Figure 2.3: Scatter plot of MPG vs. Horsepower with linear regression fit, and residual plot for Example 2.2

We then have, as for ANOVA, the total sum of squares

$$SSTO = \sum_{i=1}^{n} (Y_i - \bar{Y})^2.$$

By convention, instead of the *treatment sum of squares* SST we define the *regression sum of squares*

$$SSR = \sum_{i=1}^{n} (\hat{Y}_i - \bar{Y})^2,$$

noting that the two quantities serve similar functions. It can be shown that

$$SSTO = SSR + SSE.$$

This means that, as in ANOVA, the total variation $SSTO$ can be decomposed into variation SSR explained by the model and variation SSE attributable to the error terms ϵ_i. The ANOVA table for simple linear regression therefore looks like:

Source	SS	df	MS	
Regression	SSR	1	$MSR = \frac{SSR}{1}$	$F = \frac{MSR}{MSE}$
Error	SSE	$n-2$	$MSE = \frac{SSE}{n-2}$	
Total	SSTO	$n-1$		

As in ANOVA, F has an F-distribution with 1 numerator and $n-2$ denominator degrees of freedom under the hypothesis

$$H_o : \beta_1 = 0.$$

A quantity of considerable importance is the *coefficient of determination*

$$R^2 = 1 - \frac{SSE}{SSTO} = \frac{SSR}{SSTO}$$

which can be interpreted as the proportion of the total variation explainable by the model. We always have $0 \leq R^2 \leq 1$, so that larger values (say, $R^2 \geq 0.25$) mean that the predictor X has significant explanatory power.

2.3 The Relationship Between Linear Regression and Correlation

It is important to note the similarity between the definition of r and the estimate of the slope β_1 for simple linear regression:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2}.$$

This means r and $\hat{\beta}_1$ have a close relationship:

$$\begin{aligned} r &= r\frac{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}} \\ &= \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} \times \frac{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}}{\sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}} \\ &= \hat{\beta}_1 \sqrt{\frac{S_X^2}{S_Y^2}} \end{aligned} \tag{2.2}$$

where S_X^2 and S_Y^2 are the samples variances of the X_i's and Y_i's.

When deducing the distribution properties of r, it is usually assumed that X and Y together possess a *bivariate normal distribution*. This means that X and Y are both normally distributed, and also possess a linear relationship of the form

$$Y = \beta_0 + \beta_1 X + \epsilon \tag{2.3}$$

where β_0 and β_1 are constants and $\epsilon \sim N(0, \sigma^2)$ is independent of both X and Y. It can be shown that when (2.3) holds the correlation between X and Y is

$$\rho = \rho_{XY} = \beta_1 \frac{\sigma_X}{\sigma_Y},$$

which is directly comparable to (2.2) (when convenient, subscripts may be added to the symbols r or ρ to identify the relevant random variables). Of course, one important difference remains between (2.3) and the simple linear regression model, namely that for simple linear regression X is interpreted as a nonrandom predictor variable, whereas in (2.3) X is a random variable. Nonetheless, both models depend on the very specific notion of linear dependence between two variables.

It is important to note that assuming only that X and Y are normally distributed does not suffice to define the bivariate normal distribution. The assumption of a linear relationship is also needed.

2.4 Assumptions

The assumptions underlying simple linear regression are all implied in the model defined in (2.1):

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \;\; i = 1, \ldots, n. \tag{2.4}$$

Essentially, we assume $Y_i \sim N(\mu_i, \sigma^2)$ for some σ^2 which does not vary with the index i, where the means μ_i are given by

$$\mu_i = \beta_0 + \beta_1 X_i. \tag{2.5}$$

Finally, the responses Y_i are assumed to be independent. This is equivalent to assuming that $\epsilon_1, \ldots, \epsilon_n$ is a random sample from distribution $N(0, \sigma^2)$, and the response Y_i is given by (2.1).

Chapter 3

Linear Regression - Inference

In this section we consider in more detail inference for linear regression. We will emphasize the simple linear regression model:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \;\; i = 1, \ldots, n,$$

but the ideas can be generalized when additional predictors are added.

Usually, there is greater interest in β_1, the *slope* of the regression line, than β_0, the point on the vertical response axis intercepted by the regression line at predictor value $X = 0$ (hence β_0 is commonly known as the *intercept*). This is because the motivation for regression is usually to determine a relationship between the dependent and independent variables, and a relationship can be said to exist between them if and only if the slope β_1 is not zero.

We may consider two estimation problems. The mean response for predictor value x is

$$\mu_x = \beta_0 + \beta_1 x.$$

In principle, we may consider μ_x for any value x, even if x does not equal the value of any predictor in a given sample. However, it is usually not recommended that x be *extrapolated* beyond the range of the observed predictors. If we have some reason to set $x > \max_i X_i$ or $x < \min_i X_i$, it should be noted in any report that the resulting inference represents an extrapolation beyond the observed range of the predictor variables. Of course, the intercept β_0 is a special case of μ_x, in particular,

$$\beta_0 = \beta_0 + \beta_1 \times 0 = \mu_0,$$

however, β_1 cannot be expressed as μ_x for some x in this way.

3.1 Inference of Regression Parameters

We may define a general parameter β_i, and note that it's inference assumes a general form (we have, so far, encountered β_0 and β_1 for simple linear regression). We have seen estimates

$$\hat{\beta}_i \approx \beta_i, \;\; i = 0, 1$$

and we may add

$$\hat{\mu}_x \approx \hat{\beta}_0 + \hat{\beta}_1 x.$$

Note that the *predicted responses*

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i, \;\; i = 1, \ldots, n,$$

are special cases $\hat{Y}_i = \hat{\mu}_{X_i}$, and are commonly known as *fitted values*, since the estimated regression line passes through the points $(X_i, \hat{Y}_i)$.

Under the assumption that the error terms $\epsilon_1, \ldots, \epsilon_n$ are an independent random sample from $N(0, \sigma^2)$ for some fixed variance σ^2, we have

$$\hat{\beta}_i \sim N\left(\beta_i, \sigma^2_{\hat{\beta}_i}\right),$$

and

$$\hat{\mu}_x \sim N\left(\mu_x, \sigma^2_{\hat{\mu}_x}\right).$$

It is worth noting at this point that $\hat{\beta}_i$ and $\hat{\mu}_x$ are *unbiased* estimates of β_i and μ_x, since

$$E[\hat{\beta}_i] = \beta_i \text{ and } E[\hat{\mu}_x] \sim \mu_x,$$

(not all commonly used estimators are unbiased).

For simple linear regression, the values of $\sigma^2_{\hat{\beta}_i}$ and $\sigma^2_{\hat{\mu}_x}$ are well-known:

$$\sigma^2_{\hat{\beta}_1} = \frac{\sigma^2}{\sum_{i=1}^n (X_i - \bar{X})^2} \tag{3.1}$$

and

$$\sigma^2_{\hat{\mu}_x} = \sigma^2 \left[\frac{1}{n} + \frac{(x - \bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2}\right] \tag{3.2}$$

where we have mean value of the predictor:

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}.$$

Since $\beta_0 = \mu_0$ we can obtain directly from (3.2) the variance of $\hat{\beta}_0$ by substituting $x = 0$:

$$\sigma^2_{\hat{\beta}_0} = \sigma^2 \left[\frac{1}{n} + \frac{\bar{X}^2}{\sum_{i=1}^n (X_i - \bar{X})^2}\right]. \tag{3.3}$$

As we might expect, the values of $\sigma^2_{\hat{\beta}_i}$ and $\sigma^2_{\hat{\mu}_x}$ directly depend on error variance σ^2, which is usually unknown. Of course, we already have estimate

$$\hat{\sigma}^2 = MSE = \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n - 2} \approx \sigma^2,$$

so we replace σ^2 in (3.1), (3.2) and (3.3) with $\hat{\sigma}^2$, to obtain the *standard errors*

$$S_{\hat{\beta}_1} = \frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}}, \tag{3.4}$$

$$S_{\hat{\mu}_x} = \hat{\sigma} \sqrt{\frac{1}{n} + \frac{(x - \bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2}} \tag{3.5}$$

and

$$S_{\hat{\beta}_0} = \hat{\sigma} \sqrt{\frac{1}{n} + \frac{\bar{X}^2}{\sum_{i=1}^n (X_i - \bar{X})^2}} \tag{3.6}$$

3.1.1 Confidence Intervals for Simple Linear Regression

Given the standard errors for β_i or μ_x a level $(1-\alpha)$ confidence interval for β_i is given by

$$\hat{\beta}_i \pm t_{n-2,\alpha/2} \times S_{\hat{\beta}_i},$$

or for μ_x by

$$\hat{\mu}_x \pm t_{n-2,\alpha/2} \times S_{\hat{\mu}_x},$$

where $t_{n-2,\alpha/2}$ is the $\alpha/2$ critical value for a t-distribution with $n-2$ degrees of freedom.

3.1.2 Hypothesis Tests for Simple Linear Regression

If we wish to test against a hypothesis

$$H_o : \beta_i = \beta_i' \tag{3.7}$$

we use statistic

$$T = \frac{\hat{\beta}_i - \beta'}{S_{\hat{\beta}_i}}$$

which, under the hypothesis defined in Equation (3.7) has a t-distribution with $n-2$ degrees of freedom.

The most common hypothesis test in the context of simple linear regression is obtained by setting hypothetical value $\beta_1' = 0$, that is, the two-sided test:

$$H_o : \beta_1 = 0 \text{ against } H_a : \beta_1 \neq 0,$$

which gives observed significance level

$$\alpha_{obs} = 2P\left(T \leq -|T_{obs}|\right)$$

where

$$T_{obs} = \frac{\hat{\beta}_1}{S_{\hat{\beta}_1}}$$

and T has a t-distribution with $n-2$ degrees of freedom. When suitable, one-sided hypothesis tests can be carried out as descussed in previous chapters.

3.1.3 Prediction Intervals for Simple Linear Regression

Let's define a random variable

$$Y_x \sim N(\mu_x, \sigma^2),$$

which can be interpreted as a *future response* from a linear model

$$Y_x = \beta_0 + \beta_1 x + \epsilon$$

for a given predictor value $X = x$, and $\epsilon \sim N(0, \sigma^2)$. We might with to place *prediction bounds* on Y_x, that is, values Y_L, Y_U for which

$$P(Y_L \leq Y_x \leq Y_U) = 1 - \alpha.$$

We might set $1-\alpha = 95\%$. If $\beta_0, \beta_1, \sigma^2$ are known, this is easy to do:

$$\begin{aligned} Y_L &= \mu_x - z_{\alpha/2}\sigma, \\ Y_U &= \mu_x + z_{\alpha/2}\sigma. \end{aligned}$$

Otherwise, we estimate μ_x and σ^2, and the prediction interval can be based on the deviation

$$D = Y_x - \hat{\mu}_x,$$

that is, the deviation of a future response Y_x from it's estimated mean $\hat{\mu}_x$. At this point we note that Y_x, being some future response, is independent of the data used to estimate $\hat{\mu}_x$. This means Y_x and $\hat{\mu}_x$ are independent, so that the variance of D is

$$\begin{aligned} var(D) &= var(Y_x) + var(\hat{\mu}_x) \\ &= \sigma^2 + \sigma^2_{\hat{\mu}_x} \\ &= \sigma^2 \left[1 + \frac{1}{n} + \frac{(x-\bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2} \right] \end{aligned}$$

making use of Equation (3.2). This leads to level $(1-\alpha)$ prediction interval

$$\hat{\mu}_x \pm t_{n-2,\alpha/2} \times \hat{\sigma} \left[1 + \frac{1}{n} + \frac{(x-\bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2} \right]^{1/2}.$$

3.1.4 Calculations Based on Sums of Squares

Despite the apparent complexity of the computations associated with linear regression, they can be organized around the 5 quantities

$$\sum_{i=1}^n X_i, \ \sum_{i=1}^n Y_i, \ \sum_{i=1}^n X_i^2, \ \sum_{i=1}^n Y_i^2, \ \sum_{i=1}^n X_i Y_i$$

from which we derive quantities

$$\begin{aligned} \bar{X} &= \frac{\sum_{i=1}^n X_i}{n} \\ \bar{Y} &= \frac{\sum_{i=1}^n Y_i}{n} \\ SS_X &= \sum_{i=1}^n (X_i - \bar{X})^2 = \sum_{i=1}^n X_i^2 - n^{-1} \left(\sum_{i=1}^n X_i \right)^2 \\ SS_Y &= \sum_{i=1}^n (Y_i - \bar{Y})^2 = \sum_{i=1}^n Y_i^2 - n^{-1} \left(\sum_{i=1}^n Y_i \right)^2 \\ SS_{XY} &= \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y}) = \sum_{i=1}^n X_i Y_i - n^{-1} \left(\sum_{i=1}^n X_i \right) \left(\sum_{i=1}^n Y_i \right). \end{aligned}$$

These quantities appeared in Section 4 as $\bar{X}_n, \bar{Y}_n, SS_X(n), SS_Y(n), SS_{XY}(n)$, but we omit reference to sample size n here for convenience.

The relevant quantities then become

$$\begin{aligned}\hat{\beta}_1 &= \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} = \frac{SS_{XY}}{SS_X},\\ \hat{\beta}_0 &= \bar{Y} - \hat{\beta}_1 \bar{X},\\ \hat{\mu}_x &= \hat{\beta}_0 + \hat{\beta}_1 x.\end{aligned}$$

This means estimates are most conveniently calculated in the order $\hat{\beta}_1$, $\hat{\beta}_0$ and $\hat{\mu}_x$ as required.

We next calculate SSE and $SSTO$, following which we may calculate any required standard errors. First

$$SSTO = SS_Y.$$

Although the calculation of SSE is not, at first, as straightforward, it can be shown that

$$SSE = \sum_{i=1}^n Y_i^2 - \hat{\beta}_0 \sum_{i=1}^n Y_i - \hat{\beta}_1 \sum_{i=1}^n X_i Y_i.$$

giving

$$\hat{\sigma}^2 = MSE = \frac{SSE}{n-2}$$

and coefficient of determination

$$R^2 = 1 - \frac{SSE}{SSTO}.$$

At this point we may calcuate standard errors:

$$S_{\hat{\beta}_1} = \frac{\hat{\sigma}}{\sqrt{SS_X}},$$

$$S_{\hat{\mu}_x} = \hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x-\bar{X})^2}{SS_X}}$$

and

$$S_{\hat{\beta}_0} = \hat{\sigma}\sqrt{\frac{1}{n} + \frac{\bar{X}^2}{SS_X}},$$

with $(1-\alpha)$ prediction interval for Y_x

$$\hat{\mu}_x \pm t_{n-2,\alpha/2} \times \hat{\sigma}\left[1 + \frac{1}{n} + \frac{(x-\bar{X})^2}{SS_X}\right]^{1/2}.$$

Example 3.1. The following example is due to Devore 1995 (Example 12.10). Suppose we are given data ($n = 11$):

```
X = 16.1 31.5 21.5 22.4 20.5 28.4 30.3 25.6 32.7 29.2 34.7
Y = 4.41 6.81 5.26 5.99 5.92 6.14 6.84 5.87 7.03 6.89 7.87
```

We wish to construct a CI for β_1. We have the summary

$$\begin{aligned}
\sum_{i=1}^{n} X_i &= 292.90 \\
\sum_{i=1}^{n} Y_i &= 69.03 \\
\sum_{i=1}^{n} X_i^2 &= 8141.75 \\
\sum_{i=1}^{n} Y_i^2 &= 442.1903 \\
\sum_{i=1}^{n} X_i Y_i &= 1890.200.
\end{aligned}$$

We then have

$$\begin{aligned}
\bar{X} &= 292.9/11 = 26.627 \\
\bar{Y} &= 69.03/11 = 6.275 \\
SS_X &= 8141.75 - (292.9^2)/11 = 342.622 \\
SS_Y &= 442.1903 - (69.03^2)/11 = 8.996 \\
SS_{XY} &= 1890.20 - 292.9 * 69.03/11 = 52.119,
\end{aligned}$$

Giving coefficient estimates:

$$\begin{aligned}
\hat{\beta}_1 &= \frac{SS_{XY}}{SS_X} = \frac{52.119}{342.622} = 0.152 \\
\hat{\beta}_0 &= \bar{Y} - \hat{\beta}_1 \bar{X} = 6.275 - 0.1520 \times 26.627 = 2.228.
\end{aligned}$$

The next step is to calculate SSE:

$$\begin{aligned}
SSE &= \sum_{i=1}^{n} Y_i^2 - \hat{\beta}_0 \sum_{i=1}^{n} Y_i - \hat{\beta}_1 \sum_{i=1}^{n} X_i Y_i \\
&= 442.1903 - 2.228 \times 69.03 - 0.152 \times 1890.200 \\
&= 1.08,
\end{aligned}$$

then

$$\hat{\sigma}^2 = SSE/(n-2) = 1.08/9 = 0.12.$$

Then

$$S_{\hat{\beta}_1} = \frac{\hat{\sigma}}{\sqrt{SS_X}} = \frac{\sqrt{0.12}}{\sqrt{342.622}} = 0.019.$$

A level 95% confidence interval is then

$$0.152 \pm t_{9,\alpha/2} S_{\hat{\beta}_1} = 0.152 \pm 2.262 \times 0.019 = 0.152 \pm 0.042.$$

To calculate the coefficient of determination we set

$$SSTO = SS_Y = 8.992$$

so that

$$R^2 = 1 - SSE/SSTO = 1 - 1.08/8.992 = 0.88.$$

The high value for R^2 is evident in Figure 3.1. □

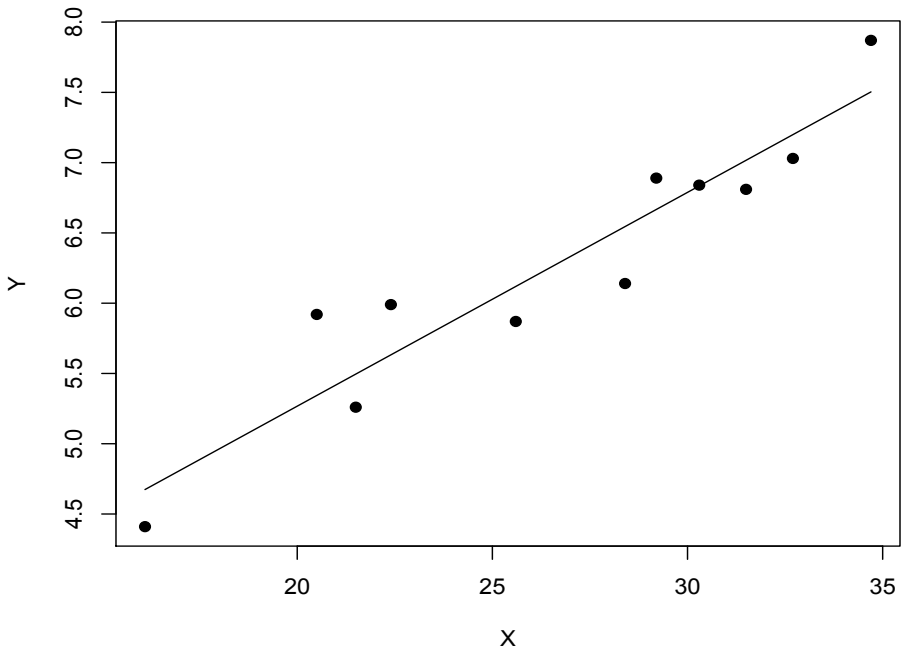

Figure 3.1: Scatter plot and regression fit for Example 3.1

3.2 Multiple Linear Regression

In contrast with simple regression, *multiple regression* permits $q \geq 1$ predictors:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_1 X_{2i} + \ldots + \beta_q X_{qi} + \epsilon_i, \;\; i = 1, \ldots, n. \tag{3.8}$$

The predictors X_{ji} use a double subscript notation, where j refers to the predictor, and i refers to the sample. So, instead of observations in pairs (Y_i, X_i) for simple linear regression, observations come in the form $(Y_i, X_{1i}, X_{2i}, \ldots, X_{qi})$ for $i = 1, \ldots, n$. In the context of multiple regression, it is usually the practice to refer to the jth predictor as X_j, on the understanding that a second subscript is needed to refer to the actual data. As in simple linear regression, the error terms $\epsilon_1, \ldots, \epsilon_n$ are a random sample from $N(0, \sigma^2)$, so that

$$Y_i \sim N(\mu_i, \sigma^2)$$

where

$$\mu_i = \beta_0 + \beta_1 X_{1i} + \beta_1 X_{2i} + \ldots + \beta_q X_{qi}, \;\; i = 1, \ldots, n$$

defines the *linear regression function.*

For each coefficient β_i we may obtain a *least squares estimate* $\hat{\beta}_i$ and standard error $S_{\hat{\beta}_i}$. Their calculation requires techniques from matrix algebra which are beyond the scope of this course, so we rely on statistical computing. As in simple linear regression we have predicted, or fitted, values

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{1i} + \hat{\beta}_1 X_{2i} + \ldots + \hat{\beta}_q X_{qi}, \;\; i = 1, \ldots, n,$$

residuals

$$e_i = Y_i - \hat{Y}_i, \;\; i = 1, \ldots, n,$$

error sum of squares

$$SSE = \sum_{i=1}^{n} e_i^2,$$

and total sum of squares

$$SSTO = SS_Y = \sum_{i=1}^{n} (Y_i - \bar{Y})^2.$$

The regression sum of squares is similarly obtained from the equality

$$SSTO = SSR + SSE.$$

We also have the various mean sums of squares. The degrees of freedom associated with SSTO remains $n - 1$, but for SSE it is now $n - (q + 1)$, and for SSR it is q, giving

$$\begin{aligned} MSE &= \frac{SSE}{n - (q+1)}, \\ MSR &= \frac{SSR}{q}. \end{aligned}$$

As in the simple linear regression case, and estimate of the error variance σ^2 is given by

$$\sigma^2 \approx \hat{\sigma}^2 = MSE.$$

Confidence intervals for each coefficient β_j are given by

$$\hat{\beta}_j \pm t_{n-(q+1),\alpha/2} \times S_{\hat{\beta}_j}.$$

A test against null hypothesis

$$H_o : \beta_j = \beta_j'$$

can be based on test statistic

$$T = \frac{\hat{\beta}_j - \beta'}{S_{\hat{\beta}_j}},$$

which under H_o has a t-distribution with $n - (q + 1)$ degrees of freedom. If the null hypothesis $H_o : \beta_j = 0$ can be rejected, we may conclude that the response depends on the jth predictor X_j (in addition, possibly, to other predictors). Otherwise, there is no relationship between X_j and the response, and this predictor need not be included in the model (we usually include β_0 in the model without the need of any formal inference).

3.2.1 ANOVA tables for multple linear regression

The ANOVA table extends naturally to the multple linear regression case:

Source	SS	df	MS	
Regression	SSR	q	$MSR = \frac{SSR}{q}$	$F = \frac{MSR}{MSE}$
Error	SSE	$n-(q+1)$	$MSE = \frac{SSE}{n-(q+1)}$	
Total	SSTO	$n-1$		

Here F has an F-distribution with q numerator and $n-(q+1)$ denominator degrees of freedom under the hypothesis

$$H_o : \beta_i = 0, i = 1, \ldots, q.$$

Note that this hypothesis does not specify that the intercept β_0 is 0.

It the context of multiple regression the *coefficient of multiple determination* is

$$R^2 = 1 - \frac{SSE}{SSTO} = \frac{SSR}{SSTO}.$$

This definition is equivalent to the coefficient of determination defined for simple linear regression, but the alternative terminology emphasizes the influence on R^2 of the number of parameters in the model.

3.2.2 Full and Reduced Models

Before we consider an actual example, it is important to understand the concept of the *full and reduced models*. Suppose we begin with the model (3.8) with q predictors. If any coefficient β_i is actually 0, there is no need to include it in the model. Of course, we don't know the exact value of β_i, but we might conclude on a statistical basis that it is not signficantly different from 0, and so on that basis we can decide which predictors to keep in the model. It might seem that all we need to do is test each coefficient separately, keeping only those for which the null hypothesis $H_o : \beta_i = 0$ is rejected. There are two concerns with this approach. First, two predictors may be correlated with each other. When this happens, the respective coefficients may become difficult to interpet independently. In this case it is better to assess the predictive ability of the model as a whole. In addition, separate inferences formally require multiple testing procedures, the application of which can be cumbersome when used to select predictors for inclusion.

One basic tool for *model selection* (that is, the problem of deciding which predictors to retain in a model) is the F-test for groups of predictors. We'll refer to (3.8) as the *full model* (in a more compact form)

$$Y = \beta_0 + \beta_1 X_1 + \beta_1 X_2 + \ldots + \beta_q X_q + \epsilon \text{ Full Model.} \tag{3.9}$$

For some $p < q$ we have the *reduced model*

$$Y = \beta_0 + \beta_1 X_1 + \beta_1 X_2 + \ldots + \beta_p X_p + \epsilon \text{ Reduced Model,} \tag{3.10}$$

that is, the reduced model is obtained from the full model by removing the final $q-p$ predictors $X_{p+1}, \ldots, X_q$. We say such models are *nested models*. The motivation here is to determine whether or not the predictive ability of the reduced model can be improved by adding these final predictors

(there may be more than one of these). Of course, we have assumed that the predictors have been indexed appropriately.

Concluding that the full model is more predictive than the reduced model is equivalent to rejecting the hypothesis

$$H_o : \beta_{p+1} = \beta_{p+2} = \ldots = \beta_q = 0$$

in favor of

$$H_a : \text{ at least one of } \beta_{p+1}, \beta_{p+2}, \ldots, \beta_q \text{ is not zero.}$$

The relevant F statistic is

$$F = \frac{(SSE_p - SSE_q)/(q-p)}{SSE_q/(n-(q+1))}$$

Where SSE_q and SSE_p are the error sums of squares of the full and reduced model respectively. Under H_o F has an F-distribution with $q-p$ numerator degrees of freedom and $n-(q+1)$ denominator degrees of freedom, and so can be rejected at significance level α if

$$F_{obs} \geq F_{q-p,n-(q+1),\alpha}.$$

It is important to note that we may set $p = 0$, in which case the reduce model is simply

$$Y = \beta_0 + \epsilon,$$

that is, responses are a random sample from $N(\beta_0, \sigma^2)$ and are not related to any of the predictors (this is why β_0 is usually retained in the model). In fact, the F-statistic $F = MSR/MSE$ given in the ANOVA table is the relevant test statistic, that is, it tests against the null hypothesis

$$H_o : \beta_1 = \beta_2 = \ldots = \beta_q = 0,$$

as we have already seen.

The coefficient of multiple determination R^2 must always be interpreted carefully, since it may be shown that its value is always larger for a full model than for a (nested) reduced model, even when the relevant coefficients are truly zero. This gives the often false impression that appending a new predictor to a model improves its predictive ability. Whether or not an increase in R^2 is truly significant can be resolved by the appropriate F-test.

For this reason, we often use instead the *adjusted* R^2:

$$R^2_{adj} = 1 - \frac{SSE/(n-(q+1))}{SSTO/(n-1)}.$$

This value, in a sense, is adjusted for the number of parameters, and permits a more accurate comparison between models with differing numbers of parameters, which need not be nested.

3.2.3 Example

Consider the following output for two regression models involving independent variables

birthwt (weight of infant at birth)

headcirc (head circumference of infant at birth)

length (length of infant at birth)

toxemia (= 1 if toxins present in blood, = 0 otherwise)

The objective is to estimate an infants prenatal or neonatal weight based on various measurements, which would be observable with a sonogram. The full model would be

$$birthwt = \beta_0 + \beta_1 \times headcirc + \beta_2 \times length + \beta_3 \times toxemia + \epsilon,$$

which has $SSE(full) = 1647237.79$ with $q = 3$ predictors. There is also a reduced model

$$birthwt = \beta_0 + \beta_1 \times headcirc + \epsilon,$$

with $SSE(reduced) = 2611443.88$ with $p = 1$ predictors. The sample size was $n = 100$. Model summaries are given below.

To test hypothesis

$$H_o : \beta_2 = \beta_3 = 0$$

against

$$H_a : \text{ at least one of } \beta_2, \beta_3 \text{ is not zero}$$

we use F-statistic

$$\begin{aligned} F &= \frac{(SSE(reduced) - SSE(full))/2}{SSE(full)/(100-4)} \\ &= \frac{(2611443.88 - 1647237.79)/2}{1647237.79/(100-4)} \\ &= 28.097. \end{aligned}$$

Under the null distribution H_o, F has an F distribution with numerator and denominator degrees of freedom $\nu_{num} = q - p = 2$ and $\nu_{den} = n - (q+1) = 96$. The p-value is very small, say $P < 0.001$, since we have critical value $F_{2,96,0.001} = 7.43$. So, the full model is more predictive than the reduced model.

Summary for reduced model:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
headcirc	1	4605298.87	4605298.87	172.82	0.0000
Residuals	98	2611443.88	26647.39		

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-1154.1087	172.1523	-6.70	0.0000
headcirc	85.1780	6.4793	13.15	0.0000

Summary for full model:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
headcirc	1	4605298.87	4605298.87	268.39	0.0000
length	1	889039.76	889039.76	51.81	0.0000
toxemia	1	75166.33	75166.33	4.38	0.0390
Residuals	96	1647237.79	17158.73		

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-1567.6048	148.7759	-10.54	0.0000
headcirc	48.3632	7.4349	6.50	0.0000
length	38.0639	5.2565	7.24	0.0000
toxemia	-67.9216	32.4518	-2.09	0.0390

Chapter 4

Linear Regression - Modeling in R

The chapter has two objectives. The first is to introduce R as a tool for statistical modeling. While this is carried out using linear regression, many of the methods are equally applicable to most other types of statistical models that one would encounter in an intermediate course on statistical methodology. For this reason, this chapter also introduces, mainly by example, some new modeling techniques which are of interest on their own.

4.1 Statistical Models

In a *statistical model* a random response Y is dependent on *predictors* $X_1, X_2, \ldots, X_m$, in the sense that the distribution of Y depends on $X_1, \ldots X_k$. In many frequently used models, the relationship is given by

$$Y = \mu(X_1, X_2, \ldots, X_m) + \epsilon, \;\; \epsilon \sim N(0, \sigma^2), \tag{4.1}$$

or equivalently,

$$Y \sim N\left(\mu(X_1, X_2, \ldots, X_m), \sigma^2\right).$$

ANOVA is a simple example of this. There is a single predictor X, which is a *factor*, or categorical variable, which assumes levels $1, \ldots k$ (that is, there are k treatments, or groups). In this case, there are k means $\mu_1, \ldots, \mu_k$, so that

$$\begin{aligned} Y &= \mu(X) + \epsilon \\ &= \mu_X + \epsilon \\ &= \mu_i + \epsilon \text{ if } X = i. \end{aligned}$$

Linear regression is a somewhat more complex example, but also conforms to Equation (4.1):

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_m X_m + \epsilon. \tag{4.2}$$

4.2 ANOVA as a Model in R

R supports a specialized notation for statistical models, based on the `formula` class. We have already seen a number of examples. In the following script, a data set consisting of a numerical vector `color.value` and a character vector `color.type` is created. There are 26 records, with

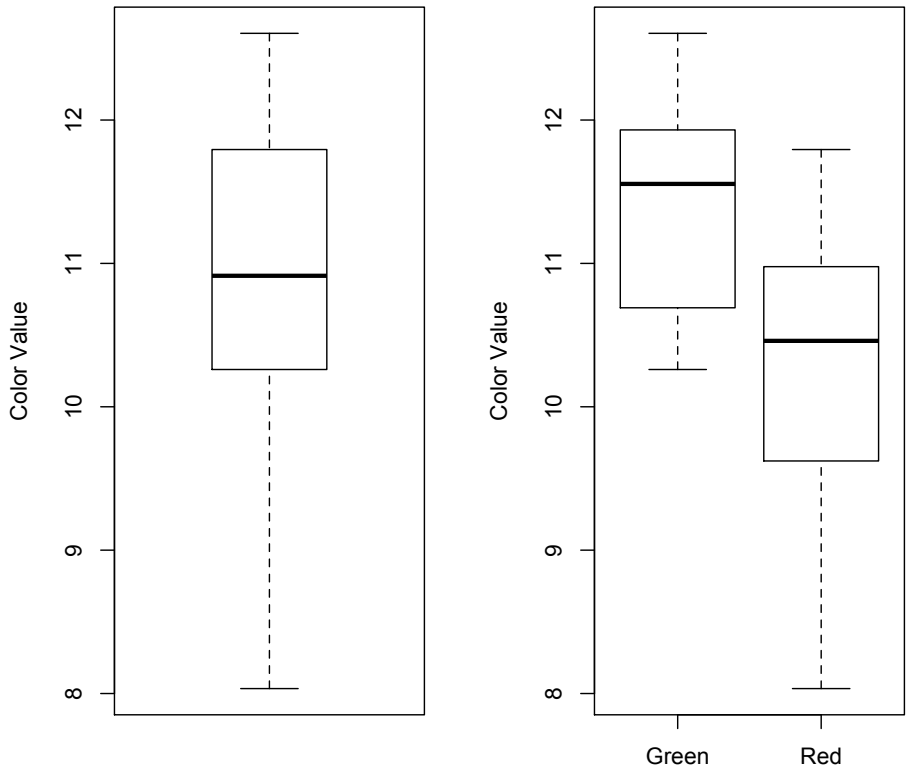

Figure 4.1: Example of the use of the `formula` class in the `boxplot`

equal numbers of "Red" and "Green" color types. The intention is that "Green" types tend to have higher values. The resulting plot is shown in Figure 4.1.

```
> par(mfrow=c(1,2),cex=1.0)
> color.value = rnorm(26,mean=rep(c(10,12),13))
> color.type = rep(c("Red","Green"),13)
> boxplot(color.value,ylab='Color Value')
> boxplot(color.value ~ color.type,ylab='Color Value')
```

The command `boxplot(color.value,ylab='Color Value')` creates a single boxplot of all the data, while the command `boxplot(color.value color.type,ylab='Color Value')` creates side-by-side boxplots for each color type.

The expression `color.value color.type` within the final boxplot command is an example of a `formula`, which takes the general form

$$\text{response} \sim \text{predictor expression}$$

It's exact effect depends on the context. In it's simplest form, as in the boxplot example of Figure 4.1, it separates a set of measurements by a group variable. However, it can also describe an analytical relationship between the response and multiple predictors.

The following script demonstrates the creation of a `formula` object, with the symbol ~ separating the response from the predictors:

```
> f1 = y ~ x
> f1
y ~ x
> class(f1)
[1] "formula"
```

The multiple regression model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \tag{4.3}$$

would be represented

```
> f1 = y ~ x1 + x2
> f1
y ~ x1 + x2
```

We will make use later in the chapter of the *interaction term*, which is a predictor formed by taking the product of two or more other predictor terms. When interactions are present in a model, the predictors forming the interaction are referred to as *main effects*. In Equation (4.3) there is one possible interaction term X_1X_2, leading to regression equation

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2.$$

To specify all possible interactions and main effects involving two predictors the * operator can be used:

```
> f2 = y ~ x1 * x2
> f2
y ~ x1 * x2
> terms(f2)
y ~ x1 * x2
attr(,"variables")
list(y, x1, x2)
attr(,"factors")
   x1 x2 x1:x2
y   0  0     0
x1  1  0     1
x2  0  1     1
attr(,"term.labels")
[1] "x1"    "x2"    "x1:x2"
attr(,"order")
[1] 1 1 2
attr(,"intercept")
```

```
[1] 1
attr(,"response")
[1] 1
attr(,".Environment")
<environment: R_GlobalEnv>
```

Note that the function `terms()` is used to extract details of a formula.

If we wanted to include only the main effect for X_1 and the interaction term, for example:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_1 X_2$$

we could use the operator `:` to generate only the specified interactions.

```
> f3 = y ~ x1 + x1:x2
> f3
y ~ x1 + x1:x2
> terms(f3)
y ~ x1 + x1:x2
attr(,"variables")
list(y, x1, x2)
attr(,"factors")
   x1 x1:x2
y   0     0
x1  1     2
x2  0     1
attr(,"term.labels")
[1] "x1"    "x1:x2"
attr(,"order")
[1] 1 2
attr(,"intercept")
[1] 1
attr(,"response")
[1] 1
attr(,".Environment")
<environment: R_GlobalEnv>
```

The intercept term is implicitly included in a formula (but can be removed if needed). If we want to model to include only the intercept (for example, for model comparisons), we use the symbol `1`:

```
 > f4 = y ~ 1
> f4
y ~ 1
> terms(f4)
y ~ 1
attr(,"variables")
list(y)
```

```
attr(,"factors")
integer(0)
attr(,"term.labels")
character(0)
attr(,"order")
integer(0)
attr(,"intercept")
[1] 1
attr(,"response")
[1] 1
attr(,".Environment")
<environment: R_GlobalEnv>
```

This is a fairly in-depth topic, so we will discuss just the basics at first (see `help(formula)` for more detail).

4.3 Linear Regression in R

First, we note that R comes with a set of example datasets, in a package called 'MASS'

```
> library(MASS)
> help(package=MASS)
...

Functions and datasets to support Venables and Ripley,
    'Modern Applied Statistics with S' (4th edition, 2002).
...
```

One of these datasets is called `nlschools`:

```
> help(nlschools)
Description

Snijders and Bosker (1999) use as a running example a study of
2287 eighth-grade pupils (aged about 11) in 132 classes in 131
schools in the Netherlands. Only the variables used in our
examples are supplied.

Usage

nlschools
Format

This data frame contains 2287 rows and the following columns:

lang
```

```
language test score.

IQ
verbal IQ.

class
class ID.

GS
class size: number of eighth-grade pupils recorded in the class
(there may be others: see COMB, and some may have been omitted
with missing values).

SES
social-economic status of pupil's family.

COMB
were the pupils taught in a multi-grade class (0/1)? Classes
which contained pupils from grades 7 and 8 are coded 1,
but only eighth-graders were tested.

Source

Snijders, T. A. B. and Bosker, R. J. (1999) Multilevel Analysis.
An Introduction to Basic and Advanced Multilevel Modelling.
London: Sage.

References

Venables, W. N. and Ripley, B. D. (2002) Modern Applied
Statistics with S. Fourth edition. Springer.
```

The object `nlschools` is a data frame. You can verify that it is a list (a data frame is a list) using the `is.list()` function. The names of the variables is obtained using the `names()` command.

```
> is.list(nlschools)
[1] TRUE
> names(nlschools)
[1] "lang"  "IQ"    "class" "GS"    "SES"   "COMB"
> nlschools[1:5,]
  lang   IQ class GS SES COMB
1   46 15.0   180 29  23    0
2   45 14.5   180 29  10    0
3   33  9.5   180 29  15    0
4   46 11.0   180 29  23    0
5   20  8.0   180 29  10    0
```

```
> dim(nlschools)
[1] 2287    6
>
```

There are 2287 rows and 6 columns.

Note that in the dataset `nlschols`, `COMB` is an *indicator variable*, that is, a variable that assumes only values $0, 1$ (or, a factor with two levels). We can do a t-test to see if there is a significant difference in language test scores between students in multigrade classes, and those not in multigrade classes. We can use model notation, but we need to specify the data frame with the `data` option.

```
> t.test(lang ~ COMB, data=nlschools)

Welch Two Sample t-test

data:  lang by COMB
t = 5.3849, df = 991.978, p-value = 9.052e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.540178 3.306355
sample estimates:
mean in group 0 mean in group 1
       41.60133        39.17806
```

So, for `COMB = 0` (not in multigrade class) the estimated mean score is $\hat{\mu}_0 = 41.6 \approx \mu_0$ and for `COMB = 1` (in multigrade class) the estimated mean score is $\hat{\mu}_1 = 39.2 \approx \mu_1$. There is a significant detrimental effect (about 2.4 points) on language test scores attributable to presence in mutligrade class.

Next, suppose we consider regression model

$$lang = \beta_0 + \beta_1 \times COMB + \epsilon$$

Since `COMB` is an indicator variable, we can match the regression coefficients directly to the two group means:

$$\begin{aligned} \mu_0 &= \beta_0 \\ \mu_1 &= \beta_0 + \beta_1, \text{ with estimates} \\ \hat{\mu}_0 &= \hat{\beta}_0 \\ \hat{\mu}_1 &= \hat{\beta}_0 + \hat{\beta}_1. \end{aligned}$$

In `R`, regression fits can be calculated using the `lm()` function, using the model notation:

```
> fit = lm(lang ~ COMB, data=nlschools)
> summary(fit)

Call:
lm(formula = lang ~ COMB, data = nlschools)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-30.1781  -6.1781   0.8219   7.3987  18.8219

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  41.6013     0.2196 189.472  < 2e-16 ***
COMB1        -2.4233     0.4187  -5.788  8.1e-09 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 8.94 on 2285 degrees of freedom
Multiple R-squared:  0.01445,Adjusted R-squared:  0.01402
F-statistic:  33.5 on 1 and 2285 DF,  p-value: 8.1e-09
```

We have coefficient estimates $\hat{\beta}_0 = 41.6013$ and $\hat{\beta}_1 = -2.4233$, giving

$$\begin{aligned} \hat{\mu}_0 &= 41.6013 \\ \hat{\mu}_1 &= 41.6013 - 2.4233 = 39.178, \end{aligned}$$

which conform to the estimates we obtained above.

4.4 ANOVA and Linear Regression

This example is due to Johnson and Bhattacharya (Statistics: Principles and Methods, Wiley, 3rd edition). In an effort to improve the quality of recording tapes, the effects of four kinds of coatings A, B, C and D on reproduction quality are assessed by applying each to a separate sample of tape and measuring the resulting distortion. The results are given in the following table.

Coating	Sample	Sample Mean	Sum of Squares
A	10, 15, 8, 12, 15	$\bar{y}_1 = 12$	$\sum_{i=1}^{5}(y_{1i} - \bar{y}_1)^2 = 38$
B	14, 18, 21, 15	$\bar{y}_2 = 17$	$\sum_{i=1}^{4}(y_{2i} - \bar{y}_2)^2 = 30$
C	17, 16, 14, 15, 17, 15, 18	$\bar{y}_3 = 16$	$\sum_{i=1}^{7}(y_{3i} - \bar{y}_3)^2 = 12$
D	12, 15, 17, 15, 16, 15	$\bar{y}_4 = 15$	$\sum_{i=1}^{6}(y_{4i} - \bar{y}_4)^2 = 14$

We can create a data frame for the data in the following way:

```
> y1 = c(10, 15, 8, 12, 15)
> y2  = c(14, 18, 21, 15)
> y3 = c(17, 16, 14, 15, 17, 15, 18)
> y4 = c(12, 15, 17, 15, 16, 15)
> y = c(y1,y2,y3,y4)
> gr = c(rep("A",5), rep("B",4), rep("C",7), rep("D",6) )
>
```

```
> tapes.data = data.frame(y,gr)
> tapes.data
    y gr
1  10  A
2  15  A
3   8  A
4  12  A
5  15  A
6  14  B
7  18  B
8  21  B
9  15  B
10 17  C
11 16  C
12 14  C
13 15  C
14 17  C
15 15  C
16 18  C
17 12  D
18 15  D
19 17  D
20 15  D
21 16  D
22 15  D
>
```

The variable y contains the responses, with treatment groups indicators by the factor variable gr. As in the previous example, we can express the ANOVA model as a linear regression model using indicator variables:

$$Y = \beta_0 + \beta_1 \times I_B + \beta_2 \times I_C + \beta_3 \times I_D + \epsilon$$

where, for example, I_B is the indicator variable for treatment B. Note that we don't need (or want) an indicator variable for treatment A. The coefficients can be related to the treatment means in the following way:

$$\begin{aligned}
\mu_A &= \beta_0 \\
\mu_B &= \beta_0 + \beta_1 \\
\mu_C &= \beta_0 + \beta_2 \\
\mu_D &= \beta_0 + \beta_3, \text{ with estimates} \\
\hat{\mu}_A &= \hat{\beta}_0 \\
\hat{\mu}_B &= \hat{\beta}_0 + \hat{\beta}_1 \\
\hat{\mu}_C &= \hat{\beta}_0 + \hat{\beta}_2 \\
\hat{\mu}_D &= \hat{\beta}_0 + \hat{\beta}_3.
\end{aligned}$$

As can be seen, we only need indicators variables for 3 of the 4 treatments. To implement this using $lm()$, we could construct the indicator variables, but the same effect can be achieved using a factor variable.

```
> fit = lm(y ~ gr, data=tapes.data)
> summary(fit)

Call:
lm(formula = y ~ gr, data = tapes.data)

Residuals:
   Min     1Q Median     3Q    Max
 -4.00  -1.75   0.00   1.00   4.00

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   12.000      1.022  11.742 7.16e-10 ***
grB            5.000      1.533   3.262  0.00433 **
grC            4.000      1.338   2.989  0.00787 **
grD            3.000      1.384   2.168  0.04381 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 2.285 on 18 degrees of freedom
Multiple R-squared:  0.4198,Adjusted R-squared:  0.323
F-statistic:  4.34 on 3 and 18 DF,  p-value: 0.01814

>
```

The coefficients match the model with

$$\begin{aligned}\hat{\beta}_0 &= 12.0\\ \hat{\beta}_1 &= 5.0\\ \hat{\beta}_2 &= 4.0\\ \hat{\beta}_3 &= 3.0,\end{aligned}$$

and we can recreate the treatment mean estimates

$$\begin{aligned}\hat{\mu}_A &= \hat{\beta}_0 = 12.0\\ \hat{\mu}_B &= \hat{\beta}_0 + \hat{\beta}_1 = 12.0 + 5.0 = 17.0\\ \hat{\mu}_C &= \hat{\beta}_0 + \hat{\beta}_2 = 12.0 + 4.0 = 16.0\\ \hat{\mu}_D &= \hat{\beta}_0 + \hat{\beta}_3 = 12.0 + 3.0 = 15.0.\end{aligned}$$

In addition, the F statistic $F = 4.34$ is equivalent to that obtained by the ANOVA procedure, as is the F test for difference in means itself.

4.5 Residuals and lm()

The output of $lm()$ is a list:

```
 > names(fit)
 [1] "coefficients"  "residuals"     "effects"       "rank"
"fitted.values" "assign"       "qr"
"df.residual"   "contrasts"     "xlevels"
[11] "call"          "terms"         "model"
```

One of the components of this list is `residuals`, which is a vector of the residuals $e_i = Y_i - \hat{Y}_i$. We can, for example, examine the normality of the residuals with a normal quantile plot (Figure 4.2):

```
> qqnorm(fit$residual)
> qqline(fit$residual)
```

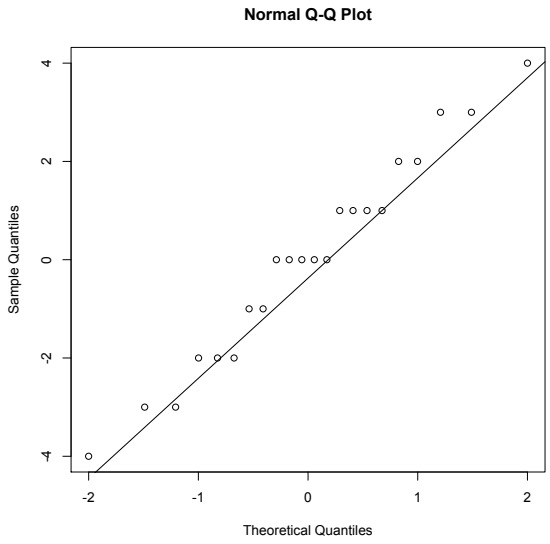

Figure 4.2: Normal quantile plot for Section 4.5.

The quantile plot suggests that the assumption of normality is reasonable.

4.6 Interaction Terms

We'll return to the `nlschools` data, and fit the model

$$lang = \beta_0 + \beta_1 \times IQ + \epsilon$$

The following script will fit the model, store the coefficients in a vector `cf`, do a scatter-plot of the independent against the dependent variable, then superimpose the actual regression line (Figure 4.3).

```
> fit = lm(lang ~ IQ, data = nlschools)
> summary(fit)

Call:
lm(formula = lang ~ IQ, data = nlschools)

Residuals:
     Min       1Q   Median       3Q      Max
-28.7022  -4.3944   0.6056   5.2595  26.2212

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.52848    0.86682   10.99   <2e-16 ***
IQ           2.65390    0.07215   36.78   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 7.137 on 2285 degrees of freedom
Multiple R-squared:  0.3719,Adjusted R-squared:  0.3716
F-statistic:  1353 on 1 and 2285 DF,  p-value: < 2.2e-16

> cf = fit$coefficients
> cf
(Intercept)          IQ
   9.528484    2.653896
> range(nlschools$IQ)
[1]  4 18
> plot(nlschools$IQ, nlschools$lang, pch=20)
> lines(range(nlschools$IQ),
cf[1] + cf[2]*range(nlschools$IQ), lwd=2)
>
```

The following script can be used to produce diagnostic plots. Note that `par(mfrow=c(1,2))` permits two plots to appear on one window. The resulting residual plot and residual normal quantile plot appear in Figure 4.4.

```
>
> par(mfrow=c(1,2))
> plot(fit$fitted.values, fit$residuals, pch=20)
> lines(c(-100,100), c(0,0))
> qqnorm(fit$residuals)
> qqline(fit$residuals)
>
```

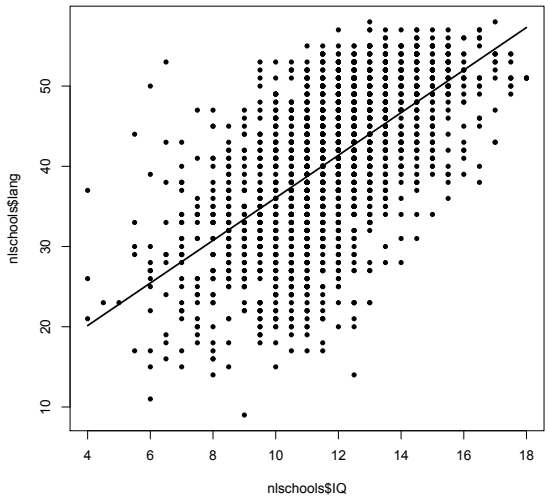

Figure 4.3: Regression fit for model $lang = \beta_0 + \beta_1 \times IQ + \epsilon$

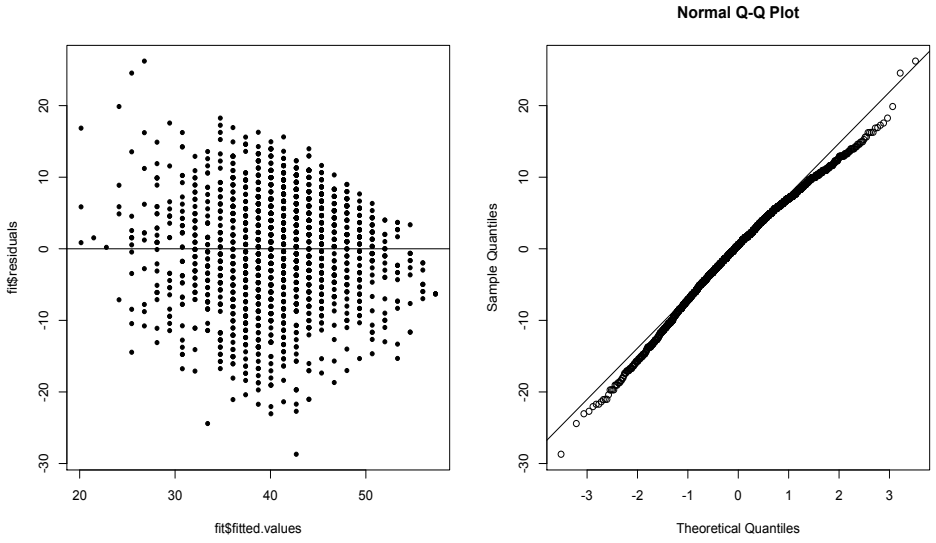

Figure 4.4: Residual plot and residual normal quantile plot for model $lang = \beta_0 + \beta_1 \times IQ + \epsilon$

Note that these plots may also be obtained using `plot(fit)`. This type of feature is generally available for models in `R`.

Next, recall that the variable `COMB` had a significant effect on the language scores, so we may wish to introduce it into out model. First, remember to change the graphics window properties if needed (here, we only want one plot, so use `par(mfrow=c(1,1))`. We now have model

$$lang = \beta_0 + \beta_1 \times IQ + \beta_2 \times COMB + \epsilon$$

where COMB is an indicator variable. However, we can consider this as two linear regression models, one for multigrade classes, and one for single grade classes. We can then superimpose two fits on one plot, in particular $y = \beta_0 + \beta_1 x$ for single grade classes, and $y = (\beta_0 + \beta_2) + \beta_1 x$ for multigrade classes.

```
> par(mfrow=c(1,1))
> fit2 = lm(lang ~ IQ + COMB, data = nlschools)
> summary(fit2)

Call:
lm(formula = lang ~ IQ + COMB, data = nlschools)

Residuals:
     Min       1Q   Median       3Q      Max
-27.3890  -4.4989   0.5011   5.1841  25.6214

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 10.25824    0.87212   11.76  < 2e-16 ***
IQ           2.63390    0.07181   36.68  < 2e-16 ***
COMB1       -1.79296    0.33265   -5.39 7.77e-08 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 7.094 on 2284 degrees of freedom
Multiple R-squared:  0.3798,Adjusted R-squared:  0.3792
F-statistic: 699.2 on 2 and 2284 DF,  p-value: < 2.2e-16

> cf = fit2$coefficients
> cf
(Intercept)          IQ       COMB1
  10.258240    2.633900   -1.792956
> plot(nlschools$IQ, nlschools$lang, pch=20)
> lines(range(nlschools$IQ),
cf[1] + cf[2]*range(nlschools$IQ), lwd=2, col=2)
> lines(range(nlschools$IQ),
cf[1] + cf[3] + cf[2]*range(nlschools$IQ), lwd=2, col=3)
> legend(14,20,
legend=c("Multigrade class", "Single grade class"),
lty=c(1,1), col=c(3,2))
>
```

Note here the use of the col option in plot() to color lines, thus distriguishing the groups. Also, the lwd option controls the width of the line, and pch defines the plotting symbol type. The legend() fuinction is then used to add a legend. Compare the magnitude of the COMB effect of -1.79296, to the COMB effect obtained by the two sample mean comparison (also obtained by the

simple regression fit $lang = \beta_0 + \beta_1 \times COMB$) of -2.4233. The resulting plot is shown in Figure 4.5.

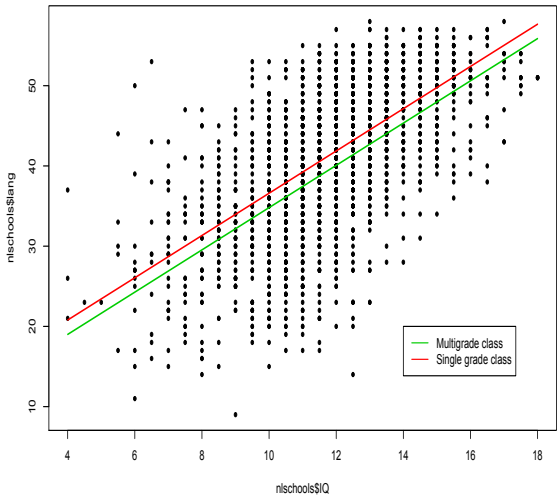

Figure 4.5: Regression fit for model $lang = \beta_0 + \beta_1 \times IQ + \beta_2 \times COMB + \epsilon$

We may wonder, however, whether or not the slope of the regression line also differs by `COMB` group. To test for this, we can use *interaction terms*, which are simply products of other predictors. These are often very useful. When interactions are present, their components are refered to as *main effects*.

For example, we might fit model:

$$lang = \beta_0 + \beta_1 \times IQ + \beta_2 \times COMB + \beta_3 \times IQ \times COMB + \epsilon.$$

Here, both the intercept and slope can differ by `COMB` group:

$$lang = \beta_0 + \beta_1 \times IQ + \epsilon, \quad \text{for } COMB = 0$$

and

$$lang = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) \times IQ + \epsilon, \quad \text{for } COMB = 1.$$

In other words, the intercepts and slopes differ between the two `COMB` groups by β_2 and β_3 respectively, therefore such differences can be tested based on null hypotheses $H_o : \beta_2 = 0$ and $H_o : \beta_3 = 0$.

We can fit this model using `lm()` (Figure 4.6). Interaction between two predictors are defined in model notion using the operator ":". Alternatively, the operator "*" will introduce interations and *main effects*.

```
> par(mfrow=c(1,1))
> fit2 = lm(lang ~ IQ + COMB + IQ:COMB, data = nlschools)
> summary(fit2)
```

```
Call:
lm(formula = lang ~ IQ + COMB + IQ:COMB, data = nlschools)

Residuals:
    Min      1Q  Median      3Q     Max
-27.768  -4.484   0.473   5.153  24.646

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  12.4068     1.0228  12.131  < 2e-16 ***
IQ            2.4533     0.0847  28.966  < 2e-16 ***
COMB1        -9.2019     1.8875  -4.875 1.16e-06 ***
IQ:COMB1      0.6317     0.1584   3.987 6.90e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 7.071 on 2283 degrees of freedom
Multiple R-squared:  0.3841,Adjusted R-squared:  0.3832
F-statistic: 474.5 on 3 and 2283 DF,  p-value: < 2.2e-16

> cf = fit2$coefficients
> cf
(Intercept)          IQ       COMB1    IQ:COMB1
  12.406772    2.453349   -9.201913    0.631680
> plot(nlschools$IQ, nlschools$lang, pch=20)
> lines(range(nlschools$IQ),
cf[1] + cf[2]*range(nlschools$IQ), lwd=2, col=2)
> lines(range(nlschools$IQ),
cf[1] + cf[3] + (cf[2]+cf[4])*range(nlschools$IQ), lwd=2, col=3)
> legend(14,20, legend=c("Multigrade class", "Single grade class"),
lty=c(1,1), col=c(3,2))
```

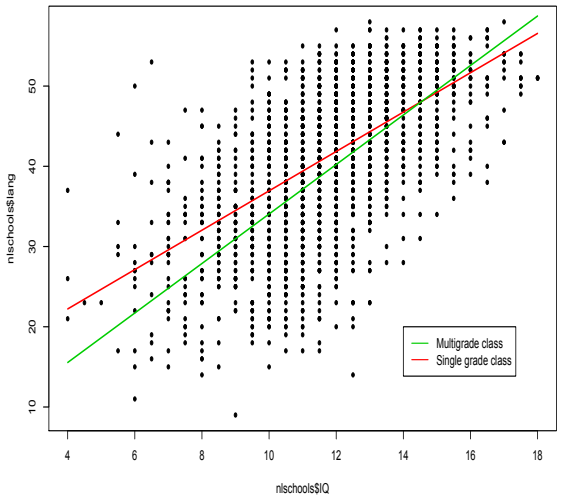

Figure 4.6: Regression fit for model $lang = \beta_0 + \beta_1 \times IQ + \beta_2 \times COMB + \beta_3 \times IQ \times COMB + \epsilon$

Here we see that language scores differ by `COMB` group, but also that this difference is larger among students with lower IQs. We say that `IQ` *interacts* with the `COMB` factor. This suggests that the performance of students with higher IQs may not be as influenced by external factors as other students are.

4.7 Polynomial Regression

The terms *linear* in *linear regression* nominally refers to the relationship between the predictors and the response. However, this has as much to do with the form of the inference as with any functional relationship. Suppose we have a model of the form

$$Y = 10 + 2.3x - 0.2x^2 + \epsilon \tag{4.4}$$

where ϵ is the familiar error term. If we regard x as a single predictor, than Equation (4.4) conforms to (4.1) but not (4.2). On the other hand, we could also regard $X_1 = x$ and $X_2 = x^2$ as two distinct predictors, in which case (4.4) conforms to both (4.1) and (4.2), with $\beta_0 = 10$, $\beta_1 = 2.3$ and $\beta_2 = -0.2$.

Fitting this type of model using multiple linear regression is referred to as *polynomial regression*. The methodology and inference remain exactly the same, as long as the linear structure of the inference is understood. In fact, introducing *quadratic terms* into a regression equation is a common method of both testing for nonlinear relationships between response and predictor, and for modeling such relationships when appropriate. We might first compare the full and reduced models

$$Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon,$$
$$Y = \beta_0 + \beta_1 x + \epsilon.$$

In this case, if a hypothesis test is able to reject the hypothesis $H_o : \beta_2 = 0$, then the full model could be accepted and summarized. For more complex problems of this type the methods of Section 3.2.2

are available. We should note that somewhat more mathematically sophisticated methods exist for polynomial regression, which are implemented in R. One common practice is to use $(x-\bar{x})^2$ instead of x^2 as the quadratic term, particularly when a large range in x leads to a much larger range in x^2. Although the fitted values $\hat{Y}$ will be identical using either form, the actual coefficient values will be different, and are usually more intuitively interpretable using the form $(x-\bar{x})^2$.

The quadratic term can be introduced into an R formula object using the notation I(x^2).

The following script simulates data from the model of Equation (4.4), using 19 equally spaced values for x ranging from 0 to 5.4. The error terms have standard deviation $\sigma = 0.5$ (how can you tell this?). The model formula is created in object lrform, and used directly in function lm(). In general, elements of a formula can refer to columns in a data frame, which would then be explicitly reference using the data option in the lm() function, as shown in the examples using the nlschools data frame earlier in this chapter.

```
> f0 = function(x) {10 + 2.3*x - 0.2*x^2}
>
> x = seq(0,5.5,0.3)
> xsq = x^2
>
> plot(x,f0(x))
>
> mux = f0(x)
> y = mux + rnorm(length(x))/2
> plot(x,y,pch=20,cex=1)
> lrform = y~x + I(x^2)
> lrform
y ~ x + I(x^2)
> fit = lm(lrform)
> summary(fit)

Call:
lm(formula = lrform)

Residuals:
     Min       1Q   Median       3Q      Max
-0.56569 -0.25812 -0.05227  0.24923  0.75776

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.73262    0.24194  40.228  < 2e-16 ***
x            2.53668    0.20771  12.212  1.6e-09 ***
I(x^2)      -0.25088    0.03713  -6.758  4.6e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.3892 on 16 degrees of freedom
Multiple R-squared:  0.9701,Adjusted R-squared:  0.9663
```

```
F-statistic: 259.4 on 2 and 16 DF,  p-value: 6.417e-13

> lines(x,mux,lty=2)
> lines(x,predict(fit),lty=1)
> legend('bottomright',legend=c('True Mean Response','Estimated Mean Response'),
       col=c(1,1),lty=c(2,1))
>
```

The resulting plot is shown in Figure 4.7. The fitted and true mean response curves are shown, along with the data points. The estimates $\hat{\beta}_0 = 9.73262$, $\hat{\beta}_1 = 2.53668$ and $\hat{\beta}_2 = -0.25088$ are quite close to the true values $\beta_0 = 10.0$, $\beta_1 = 2.3$ and $\beta_2 = -0.2$.

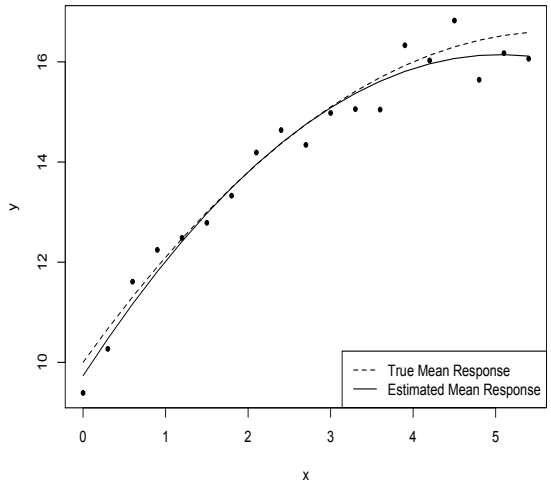

Figure 4.7: Data and linear model fit for polynomial regression example of Section 4.7.

Chapter 5

Linear Regression - Formulation Using Matrix Algebra

There will be considerable advantage to expressing linear regression in terms of linear algebra. At this point we adopt the notation of Chapter 12. The responses are $\mathbf{y}$ and the predictors are $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_q]$. What was written before as model:

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \epsilon_i, \;\; i = 1, \ldots, n, \;\; \epsilon_i \sim N(0, \sigma^2),$$

now becomes

$$\mathbf{y} = \beta_1 \mathbf{x}_1 + \ldots + \beta_q \mathbf{x}_q + \boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where $\boldsymbol{\epsilon}$ is the column vector of error terms ϵ_i. It is important to note that the intercept term has not been removed. It is now column vector $\vec{1} = [1 \cdots 1]^T$ in $\mathbf{X}$. This means if we have p predictors in the usual sense, then $\mathbf{X}$ will have $q = p + 1$ column vectors, assuming the intercept is to be included.

5.1 Regression Coefficients $\boldsymbol{\beta}$

The least squares estimates of regression coefficients $\boldsymbol{\beta}$ is given by

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{5.1}$$

using standard matrix multiplication. The covariance matrix of $\hat{\boldsymbol{\beta}}$ is given by

$$\Sigma_{\hat{\boldsymbol{\beta}}} = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}. \tag{5.2}$$

Then the estimated covariance matrix is

$$S^2_{\hat{\boldsymbol{\beta}}} = MSE \times (\mathbf{X}^T\mathbf{X})^{-1} \tag{5.3}$$

so that the standard errors referred to in Section 3.2 are given directly by

$$S_{\hat{\beta}_j} = \sqrt{MSE \times [(\mathbf{X}^T\mathbf{X})^{-1}]_{jj}}, \tag{5.4}$$

that is, the square root of the jth diagonal element of matrix (5.3).

5.2 Linear Combinations of $\boldsymbol{\beta}$

One important problem in linear regression is the estimation of linear combinations of the regression coefficients. Let

$$\eta = a_1\beta_1 + \ldots a_q\beta_q = \boldsymbol{a}^T\boldsymbol{\beta},$$

where $\boldsymbol{a} = [a_1 \cdots a_q]^T$ is the appropriate column vector. The obvious estimator for η is

$$\hat{\eta} = \boldsymbol{a}^T\hat{\boldsymbol{\beta}}. \tag{5.5}$$

It may also be shown that the standard error $S_{\hat{\eta}}$ of this estimate is given by

$$S^2_{\hat{\eta}} = MSE \times \boldsymbol{a}^T(\mathbf{X}^T\mathbf{X})^{-1}\boldsymbol{a}, \tag{5.6}$$

which may be used in the inference procedures described in Section 3.2.

5.3 Fitted Values $\hat{\boldsymbol{y}}$

The column vector of fitted values is then expressed as, using (5.1),

$$\hat{\boldsymbol{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = H\mathbf{y}$$

where

$$H = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$$

is known as the 'hat' matrix. This matrix is symmetric. It is also *idempotent*, meaning that $H = H^2$. The covariance matrix of $\mathbf{y}$ is $\Sigma_{\mathbf{y}} = \sigma^2 I_n$, where I_n is the $n \times n$ identity matrix, so the covariance matrix of $\hat{\boldsymbol{y}}$ is therefore

$$\Sigma_{\hat{\boldsymbol{y}}} = H\left[\sigma^2 I_n\right]H^T = \sigma^2 H,$$

(see Appendix B). The variance of a single fitted values is therefore

$$\sigma^2_{\hat{\boldsymbol{y}}_i} = \sigma^2 H_{ii}. \tag{5.7}$$

The standard errors are obtained by substituting MSE for σ^2:

$$\begin{aligned} S^2_{\hat{\boldsymbol{y}}} &= MSE \times H, \\ S^2_{\hat{\boldsymbol{y}}_i} &= MSE \times H_{ii}. \end{aligned} \tag{5.8}$$

5.4 Residuals $\boldsymbol{e}$

It is important to realize that while $MSE^{1/2}$ is an estimate of σ, it is not an estimate of the standard deviation of $e_i = y_i - \hat{y}_i$, which is itself a linear combination of the responses $\mathbf{y}$. Viewed this way, the residual vector is

$$\boldsymbol{e} = \mathbf{y} - \hat{\boldsymbol{y}} = [I_n - H]\mathbf{y},$$

where I_n is the $n \times n$ identity matrix. The covariance matrix of $\mathbf{y}$ is $\Sigma_{\mathbf{y}} = \sigma^2 I_n$, so the covariance matrix of $\boldsymbol{e}$ is therefore

$$\Sigma_{\boldsymbol{e}} = \sigma^2[I_n - H][I_n - H]^T = \sigma^2[I_n - H],$$

(see Appendix B) so that the variance of e_i is

$$\sigma^2_{e_i} = \sigma^2(1 - H_{ii}), \tag{5.9}$$

with standard errors

$$\begin{aligned} S^2_{\boldsymbol{e}} &= MSE \times (I_n - H), \\ S^2_{e_i} &= MSE \times (1 - H_{ii}). \end{aligned}$$

This is an important result, because it reveals that the variances of e_i are not equal, depending directly on the leverage H_{ii}. This means a high leverage observation tends to disproportinately pull the regression line towards it. In fact, since H_{ii} may be arbitrarily close to 1, from (5.9) we can see that $\sigma^2_{e_i}$ may be arbitrarily close to zero, so this effect may be quite dominant.

Chapter 6

Linear Regression Diagnostics - Outliers, Influential Observations and Collinearity

An anomalous observation in a single sample is almost always an *outlier*, or a measurement that is a relatively large distance from almost all remaining measurements. In regression, there is more than one reason to consider an observation anomalous (an observation here is collectively the response with associated predictor values). Also complicating matters is the fact that the responses are not from a single well defined distribution. Anomalies in linear regression are usually (at least) one of the following three types:

- An **outlier** is an observation with a large residual.
- An observation with **high leverage** is an observation with one or more relatively extreme predictor values.
- An observation is **influential** if its removal changes the fitted model significantly. This applies either to any of the coefficients, or to a fitted values.

Since the motivation is to test whether or not removing an observation significantly changes the model fit (relative to the remaining observations), quantitative diagnostic measures often measure the effect of deleting an observation. This can be done by simply recalculating the fit after removing each observation in turn, but explicit formula usually exist, saving considerable computation time.

We use the notation $\hat{\boldsymbol{y}}^{-i}$, $\hat{\beta}_j^{-i}$, $\left[S^2_{\hat{\boldsymbol{\beta}}}\right]^{-i}$, for example, to denote the various quantities associated with a regression model calculated after deleting the ith observation.

The reader should review Chapter 5, Appendix B, and Appendix A as needed.

6.1 Leverage

The ith diagonal element H_{ii} of the hat matrix H (Section 5.3) is referred to as the *leverage* for the ith observation. The motivation for this definition is as follows. Estimates should be reasonably stable, in the sense that a small change in a data set should not result in a large change in the

model estimate. However, if one observation has a relatively large value for H_{ii} (referred to as a *high leverage point*), this suggests that it has a disproportionately large effect on the fitted model, as can be seen by Equation (5.7). This may be problematic, since we would not like the fitted model to depend significantly on the presence or absence of one, or a few, high leverage points.

We next consider what constitutes high leverage, and there are a number of principles that may be used. The simplest approach is to examine all leverage values H_{ii} with a boxplot or histogram to detect outliers. However, there are several methods by which the magnitude of H_{ii} can be judged without having to examine all leverage values at once.

It may be shown that we always have

$$trace(H) = q,$$

where the *trace* of a square matrix is the sum of the diagonal elements. Also, it always holds that $n^{-1} \leq H_{ii} \leq 1$. If there are n observations then the average value for H_{ii} must be q/n. For this reason, high leverage points may be flagged with a simple rule such as

$$H_{ii} \geq 2q/n.$$

6.2 Cook's Distance

Another commonly used diagnostic is based on *Cook's distance*:

$$D_i = \frac{e_i^2}{q \times MSE} \left[\frac{H_{ii}}{(1 - H_{ii})^2} \right].$$

This is an interesting statistic for a number of reasons. First, it can be shown that an equivalent form for D_i is

$$D_i = \frac{\sum_{j \neq i} (\hat{y}_j - \hat{y}_j^{-i})^2}{q \times MSE},$$

where $\hat{y}_j$ is the jth fitted value using all data, and $\hat{y}_j^{-i}$ is the jth fitted value obtained after deleting observation i. The equivalence of these two forms for D_i show that a fitted model may change considerably following the addition or deletion of a high leverage point.

We also note that D_i may be compared to a $F_{q,n-q}$ distribution, and on this basis high leverage points may be flagged by comparison to an appropriate quantile. A number of rules are used, which are more or less conservative, for example:

$$D_i \geq 1.$$

6.3 Studentized Residuals

In statistics, the term *studentize* refers to the adjustment of a statistic by dividing it by its standard error, in the form of an estimate of the true standard deviation obtained from the data. The t-statistic is one example. The *studentized residuals* are therefore given by

$$e_i^* = \frac{e_i}{S'_{e_i}}, \quad i = 1, \ldots, n. \tag{6.1}$$

Because e_i^* is usually used for diagnostic purposes, we do not estimate σ_{e_i} using all the data. For this reason, we use the notation $S'_{e_i} \approx \sigma_{e_i}$ in (6.1). We retain the quantity H_{ii} appearing in the expression for $\sigma^2_{e_i}$ in (5.9). However, because we are accepting the possibility that the ith observation is anomolous, it is appropriate to use MSE^{-i} instead of MSE to estimate σ^2 in (5.9), noting that both are unbiased estimates of σ^2. This gives

$$S'_{e_i} = \sqrt{MSE^{-i}(1 - H_{ii})},$$

which, combined with (6.1) defined the studentized residual, sometimes denoted $RSTUDENT_i$.

6.4 Influence Measures

One method of determining the influence of an observation is to simply delete it, recalculate any of the various model quantities, and then note the change. Accordingly, given q predictors (including the intercept), define the quantities

$$DFBETA_{ij} = \hat{\beta}_j - \hat{\beta}_j^{-i} = \frac{\left[(\mathbf{X}^T\mathbf{X})^{-1}\dot{x}_i\right]_j e_i}{1 - H_{ii}}, \;\; i = 1, \ldots, n, \;\; j = 1, \ldots, q.$$

Similarly, define

$$DFFIT_i = \hat{\boldsymbol{y}}_i - \hat{\boldsymbol{y}}_i^{-i} = \frac{H_{ii}e_i}{1 - H_{ii}},$$

where $\hat{\boldsymbol{y}}_i^{-i}$ is the fitted value of the model at predictor value $\dot{x}_i$ recalculated after deleting the ith observation.

These quantities can be made more easily interpretable in their *standardized* form. Since $DFBETA_{ij}$ measures a change in $\hat{\beta}_j$ resulting from the deletion of the ith observation, it makes sense to standardize it by dividing by its standard error, the equation for which is given in (5.4). Since we are interested in the standard error of $\hat{\beta}_j$ and not $\hat{\beta}_j^{-i}$ (which are different) we do not delete the ith observation in $\mathbf{X}$. However, because we are accepting the possibility that the ith observation is anomolous, it is appropriate to substitute MSE^{-i} for MSE in (5.4), noting that both are unbiased estimates of σ^2. This gives the standardized form

$$DFBETAS_{ij} = \frac{DFBETA_{ij}}{\sqrt{MSE^{-i} \times [(\mathbf{X}^T\mathbf{X})^{-1}]_{jj}}}.$$

Similarly, from the standard error for $\hat{\boldsymbol{y}}_i$ given in (5.8) we have the standardized version of $DFFIT_i$:

$$DFFITS_{ij} = \frac{DFFIT_i}{\sqrt{MSE^{-i} \times H_{ii}}}.$$

6.5 Covariance Ratio

The *covariance ratio* is defined as

$$cov.ratio = \frac{\det\left([S^2_{\hat{\boldsymbol{\beta}}}]^{-i}\right)}{\det\left(S^2_{\hat{\boldsymbol{\beta}}}\right)}$$

This measures the aggregate effect of deleting an observation on the standard errors of the coefficient estimate $\hat{\boldsymbol{\beta}}$.

6.6 Collinearity

Based on (5.3), the standard error for $\hat{\beta}_j$ can be shown to be obtained from

$$S^2_{\hat{\beta}_j} = MSE \times \left[(\mathbf{X}^T\mathbf{X})^{-1}\right]_{jj} = \frac{MSE}{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2} \frac{1}{1 - R^2_{\mathbf{x}_j|\mathbf{X}^{-j}}}. \tag{6.2}$$

Here, $\bar{x}_j$ is the mean of predictor $\mathbf{x}_j$, and $R^2_{\mathbf{x}_j|\mathbf{X}^{-j}}$ is the value of R^2 obtained by regressing $\mathbf{x}_j$ onto the remaining predictors. It is interesting to compare this expression to the standard error for the slope coefficient in simple regression

$$S^2_{\hat{\beta}_1} = \frac{MSE}{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}$$

(see Equation (3.4)). The form is the same for both, except for the *variance inflation factor* (VIF):

$$VIF_j = \frac{1}{1 - R^2_{\mathbf{x}_j|\mathbf{X}^{-j}}}$$

which appears in equation (6.2). A few things are notable about VIF_j. First, the quantity $R^2_{\mathbf{x}_j|\mathbf{X}^{-j}}$ is a direct measure of *collinearity*, or the degree to which $\mathbf{x}_j$ is linearly correlated to other predictors. It can be seen that if $\mathbf{x}_j$ is exactly equal to some linear combination of other predictors, then it is not needed in the model. For example, if $\mathbf{x}_1 = \mathbf{x}_2 + \mathbf{x}_3$, then a fitted model

$$\hat{\boldsymbol{y}} = 2.5\mathbf{x}_1 - 4.7\mathbf{x}_2 + 10.4\mathbf{x}_3 \tag{6.3}$$

can always be replaced by

$$\begin{aligned} \hat{\boldsymbol{y}} &= 2.5\mathbf{x}_1 - 4.7\mathbf{x}_2 + 10.4\mathbf{x}_3 \\ &= 2.5(\mathbf{x}_2 + \mathbf{x}_3) - 4.7\mathbf{x}_2 + 10.4\mathbf{x}_3 \\ &= -2.2\mathbf{x}_2 + 12.9\mathbf{x}_3, \end{aligned} \tag{6.4}$$

and we can dispense with $\mathbf{x}_1$ entirely. Equations (6.3) and (6.4) are equivalent in the sense that they yield exactly the same fitted values. So they are the same model. However, this example does not convey the entire problem. We could easily construct a third equivalent model:

$$\hat{\boldsymbol{y}} = 1.5\mathbf{x}_1 - 3.7\mathbf{x}_2 + 11.4\mathbf{x}_3. \tag{6.5}$$

Although we generally expect the least squares estimates to uniquely minimize the SSE, when predictors are not linearly independent, there will exist an infinite number of least squares fits.

Next, suppose the predictors are linearly independent but that for some predictor $\mathbf{x}_j$, $R^2_{\mathbf{x}_j|\mathbf{X}^{-j}}$ is very close to 1. We will have a unique least squares fit, but something of the character of the preceding example remains. Models with widely varying fitted coefficients will have values of SSE close to the minimum attainable, and will yield very similar fitted values $\hat{\boldsymbol{y}}$. The consequence of

this can be seen directly, since we would then have a very large value of VIF_j and, by (6.2), a very large value for $S^2_{\hat{\beta}_j}$, meaning that the coefficient β_j cannnot be reliably estimated. A generally used rule of thumb flags collinearity effects when

$$VIF_j \geq 10.$$

Chapter 7

Maximum Likelihood Estimation

Suppose we are given a joint density $f(\tilde{X};\theta)$ of a random vector $\tilde{X} = (X_1, \ldots, X_n)$, noting that the density depends on a parameter $\theta \in \Theta$, where Θ is known as the *parameter space*. If we think of $f(\tilde{X};\theta)$ as a function on the n-dimensional sample space, fixing θ, it is a probbaility density function. However, we may also think of it as a function of θ over Θ, holding $\tilde{X}$ fixed. In this case, it is referred to as the *likelihood function*

$$l(\theta) = l(\theta;\tilde{X}) = f(\tilde{X};\theta),$$

or, equivalently, the *log-likelihood function*

$$L(\theta) = L(\theta;\tilde{X}) = \log f(\tilde{X};\theta).$$

If we are given a sample $\tilde{X}$, and we know the density has form $f(\tilde{X};\theta)$, but we don't know the value of θ, then $L(\theta;\tilde{X})$ becomes a type of index describing how well a particular parameter value $\theta' \in \Theta$ describes the data. This is because, intuitively, we would expect the likelihood $l(\theta';\tilde{X})$ to be relatively large when θ' is close to the true value of θ. Thus, the *maximum likelihood estimate* (MLE) $\hat{\theta}_{MLE}$ is defined as

$$\hat{\theta}_{MLE} = \text{argmax}_{\theta\in\Theta} L(\theta;\tilde{X}). \tag{7.1}$$

There are, of course, a few technical issues. First, there is no gaurantee that a maximum is unique, or even exists, although for many well known cases regularity conditions under which this holds have been derived. Note also that we use the log-likelihood function. It would be equivalent to use the likeihood function, but in practice the computation tends to be simpler using (7.1).

There exists a general theory that we outline here. Given a function $f(x_1, \ldots, x_n)$ of n variables the *Hessian matrix* is the matrix of second order partial derivatives:

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial^2 x_1} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial^2 x_n} \end{bmatrix}.$$

We will use the shorthand

$$H = \frac{\delta^2 f}{\delta^2 x}$$

where we use vector representation $x = (x_1, \ldots, x_n)$. The i, jth element of H is written

$$H_{ij} = \frac{\partial^2 f}{\partial x_j \partial x_j}.$$

The *Fisher information matrix* is defined as

$$\mathcal{I}(\theta) = -E\left[\frac{\delta^2 L(\theta; \tilde{X})}{\delta^2 \theta}\right]$$

and plays an important role in the theory of statistical inference. For our purposes, we note that the *observed Fisher information matrix* is simply

$$\hat{I}(\theta) = -\frac{\delta L(\theta^2; \tilde{X})}{\delta^2 \theta},$$

and it can be shown that if we set

$$\hat{I} = \hat{I}(\hat{\theta}_{MLE}),$$

then $\hat{I}^{-1}$ estimates the covariance matrix of $\hat{\theta}_{MLE}$.

This gives a general approach to formal inference in modeling. Suppose $\theta \in \Theta \subset \mathbb{R}^p$. This mean

$$\hat{\theta}_{MLE} = (\hat{\theta}_1, \ldots, \hat{\theta}_p)$$

is a p-dimensional vector. The standard error of component $\hat{\theta}_i$ is therefore

$$SE_{\hat{\theta}_i} = \sqrt{[\hat{I}^{-1}]_{ii}},$$

that is, the square root of the ith element on the diagonal of $\hat{I}^{-1}$. Then, level $1 - \alpha$ confidence intervals for the ith element of θ are given by

$$CI_{1-\alpha} = \hat{\theta}_i \pm t_{n-d;\alpha/2} SE_{\hat{\theta}_i},$$

where d is the degrees of freedom of the model, or for large sample sizes

$$CI_{1-\alpha} = \hat{\theta}_i \pm z_{\alpha/2} SE_{\hat{\theta}_i}.$$

Similarly, under a null hypothesis $H_o : \theta_i = \theta_i^*$ we have null distribution

$$\frac{\hat{\theta}_i - \theta_i^*}{SE_{\hat{\theta}_i}} \sim T_{n-d}$$

or for large samples

$$\frac{\hat{\theta}_i - \theta_i^*}{SE_{\hat{\theta}_i}} \sim N(0, 1).$$

Example 7.1. It is worth examining what the likelihood function looks like for a model we have already seen. The model for simple linerar regression is

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \;\; \epsilon_i \sim N(0, \sigma^2).$$

Usually, the error terms ϵ_i are assume to be independent. Of course, this might not be the case, but we will assume that here. This means means that the responses y_i are also independent.

How do we assign a density to this model, in order to construct a likelihood function? The first problem is to define the unknown parameters. In simple linear regression, this usually includes β_0, β_1, although even here this ultimately depends on the application. The next problem is to decide whether or not σ^2 is a parameter. It is almost always 'unknown', but if it is not the object of the inference, we may regard it as fixed (if not 'known'), taking only the regression coefficients as parameters. In this particular case, either choice will lead to the same estimated regression coefficients, so we will opt to regard σ^2 as fixed.

Note also that the predictors x_i are considered fixed. This means

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2),$$

and so the density of $\tilde{y} = (y_1, \ldots, y_n)$ is, following (B.5), is

$$f(\tilde{y}) = \prod_{i=1}^{n} \phi(y_i; \beta_0 + \beta_1 x_i, \sigma^2).$$

After some algebra, the log-likelihood function becomes

$$L(\beta_0, \beta_1; \tilde{y}) = -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 + C \tag{7.2}$$

where C is a constant which does not depend on parameters β_0, β_1, and so may be removed from the likelihood function.

Finally, examining (7.2) we can see that the estimates $\hat{\beta}_0, \hat{\beta}_1$ which maximum the log-likelihood function are exactly those that minimize the least squares criterion

$$SSE[\beta_0, \beta_1] = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2.$$

This means that the MLEs of β_0, β_1 are equal to the least squares estimates.

7.1 The Likelihood Ratio Test and Deviance

In linear regression the quantity SSE serves as a goodness of fit measure. It also serves as a means of comparing *nested models*. Suppose a full model has q predictors (in addition to the intercept). Then define a reduced model which contains only $p < q$ of these predictors. The reduced model can be considered to be a special case of the full model in which the coefficients β_i for the removed predictors are forced to zero. We then have $SSE_{red} \geq SSE_{full}$, since SSE_{full} is the minimum SSE over the full model space, which includes the reduced model. The F-statistic is then

$$F = \frac{(SSE_{red} - SSE_{full})/(q - p)}{SSE_{full}/(n - (q + 1))} \tag{7.3}$$

which has an $F_{q-p,n-(q+1)}$ distribution under tha null hypothesis that the reduced model is correct.

A similar method of inference is available for maximum likelihood estimation. Suppose we are given data $\tilde{X}$ and a likelihood function $l(\theta; \tilde{X})$ on parameter space $\Theta_f \in \mathbb{R}^q$. Then let $\Theta_r \in \mathbb{R}^p$. Assume the parameter spaces Θ_r and Θ_f are nested, in the sense that models in Θ_r are contained in Θ_f. This situation describes the comparison of full and reduced regression models just described. SImilarly, we calculate the full and reduced MLEs

$$\hat{\theta}_f = \text{argmax}_{\theta \in \Theta_f} l(\theta; \tilde{X}),$$
$$\hat{\theta}_r = \text{argmax}_{\theta \in \Theta_r} l(\theta; \tilde{X}).$$

The likelihood ratio statistic is then

$$\Lambda(\tilde{X}; \hat{\theta}_r, \hat{\theta}_f) = \frac{l(\hat{\theta}_r; \tilde{X})}{l(\hat{\theta}_f; \tilde{X})},$$

with small values of $\Lambda(\tilde{X})$ tending to support the full model. If we define null hypothesis $H_o : \theta \in \Theta_r$ (that is, the reduced model is correct) then by *Wilk's theorem*

$$-2 \log(\Lambda(\tilde{X}; \hat{\theta}_r, \hat{\theta}_f)) \sim \chi^2_{q-p} \tag{7.4}$$

approximately for large enough sample size. This serves the same purpose as the F-test defined in (7.3).

An analog of SSE may then be developed by defining the *saturated model*, for which the number of parameters is the same as the number of observations, so that the data are fitted exactly. For example, if in a linear regression model there are n linearly independent predictors (including the intercept) then we can attain $SSE = 0$ (since the fitted values can be made to equal the responses exactly). Suppose the resulting MLE is $\hat{\theta}_s$. Then any model is nested within the saturation model. Suppose $\hat{\theta}_m$ is the MLE for our model of interest. The model *deviance* is then based on the likelihood ratio test for the model of interest compared to the saturated model:

$$D(\hat{\theta}_m) = -2 \log(\Lambda(\tilde{X}; \hat{\theta}_m, \hat{\theta}_s)).$$

This statistic serves much the same purpose as the SSE, and permits a systematic comparison of models, since, for example

$$D(\hat{\theta}_r) - D(\hat{\theta}_f) = -2 \log(\Lambda(\tilde{X}; \hat{\theta}_r, \hat{\theta}_f)), \tag{7.5}$$

which is just a reformulation of (7.4). In pratice, to compare nested models, we can calculate the change in deviance (7.5), and compare this quantity to a χ^2_{q-p} distribution. If the change is large (ie. the p-value is small), then we conclude that the full model is an improvement over the reduced model.

Chapter 8

Bayesian Inference

Suppose random data X is observed, which possesses a density $f(x \mid \theta)$ from a family of models parametrized by $\theta \in \Theta \subset \mathbb{R}^p$, where Θ is known as the parameter space.

Most modeling techniques we have seen attempt to minimize prediction error. We have also seen the maximum likelihood principle. These methods can be modified to incorporate complexity penalties, but what they have in common is that the selected model is the one which optimizes some criterion.

Technically, the main difference between Bayesian inference and these other methods is that optimization is replaced by integration. In likelihood, we regard the quantity $l(\theta) = f(x \mid \theta)$ as a modeling criterion to be optimized *with respect to* θ. In Bayesian inference, θ itself is taken to be a random variable or vector. To formalize the idea, the following framework is adopted. We assume there is a *prior density* $\pi(\theta)$ for θ. This describes the range of possible values for θ, and an initial description of their relative plausibility, sometimes refered to as *belief* (or *prior belief*). This might be based on some model, or it may be entirely subjective.

We have seen exactly this form of inference before in the Bayes classifer (Chapter 13.6). Given classes $j = 1, \ldots, m$ we have prior probabilities $\pi_1, \ldots \pi_m$. In a sense we can think of class j as the parameter θ within the set of all classes $\Theta = \{1, \ldots, m\}$. We then have posterior probability, given the data

$$P(j \mid x) = \frac{f(x \mid j)\pi_j}{f(x)} = \frac{f(x \mid j)\pi_j}{\sum_{j=1}^{m} f(x \mid j)\pi_j}.$$

In other words, we have a prior distribution on the space of all models, and this distribution is altered by conditioning on data (or evidence) to yield the posterior distribution.

In much the same way, if $\pi(\theta)$ is a continuous density on a parameter space Θ in $\mathbb{R}^p$, we would have posterior density

$$\pi(\theta \mid x) = \frac{f(x \mid \theta)\pi(\theta)}{f(x)} = \frac{f(x \mid \theta)\pi(\theta)}{\int_\Theta f(x \mid \theta)\pi(\theta) d\theta}.$$

8.1 The Bayes Estimator

In Chapter 13.6, we noted that the Bayes classifier minimized classification error. A similar result holds for Bayesian inference in general.

The posterior distribution is the basis for Bayesian inference, but it is usually more convenient to refer to a single point estimate. There is a well developed theory behind this problem, which

we briefly introduce. Recall from Section 12.5 the idea of *loss* $L(x, y)$ and *risk* $R = E[L(x, y)]$. That discussion was in the context of prediction, that is, the construction of a predictor y which is meant to be close to x, based on any avaliable feature data. The ideas are much the same for estimation, in which $x = \theta$ is an unknown parameter to be estimated, and y is the estimator. The loss function serves the same purpose, and is usually taken to be squared error $L(x, y) = (x - y)^2$, although absolute deviation $L(x, y) = |x - y|$ is a commonly used alternative. In any case, we generally assume $L(x, x) = 0$. Next, suppose $\hat{\theta}$ is an estimator of θ. Risk is then

$$R(\theta, \hat{\theta}) = E_\theta[L(\theta, \hat{\theta})]$$

where the expectation is calculated assuming that θ is the correct parameter value. Any loss function may be used, but then, of course, the risk depends on that choice.

Risk is used to measure the accuracy of an estimator. We generally wish risk to be small, but we also want this to hold in some sense over the entire parameter space $\theta \in \Theta$. For example, suppose we wish to estimate θ from distribution $N(\theta, \sigma^2)$, based on observation $X \in N(\theta, \sigma^2)$. Clearly, estimator $\hat{\theta}$ cannot depend on θ, but it should depend on observation X. Suppose, ignoring this advice, we set $\hat{\theta} \equiv 10.51$ for any value of X. In fact, this would be an excellent estimator if, indeed, θ was equal to 10.5, since $R(10.51, \hat{\theta}) = 0$. But we can't expect this, and $R(\theta, \hat{\theta})$ would be very large for most θ (not near 10.51).

There are a number of ways to use risk to formulate coherent criterion for the selection of estimators. For example, suppose we have an *iid* sample from $N(\theta, \sigma^2)$. It can be shown that among unbiased estimates of θ, that is, estimators for which

$$E_\theta[\hat{\theta}] = \theta$$

for all $\theta \in \Theta$, the sample mean $\bar{X}$ has uniformly minimum squared error risk over $\theta \in (-\infty, \infty)$ (the estimator $\hat{\theta} \equiv 10.51$ is not unbiased).

Bayesian inference provides a natural method of using risk to select estimators. We first integrate risk over the prior distribution

$$B(\pi, \hat{\theta}) = \int_{\theta \in \Theta} R(\theta, \hat{\theta}) \pi(\theta) d\theta,$$

a quantity known as *Bayes risk*. We interpret $\hat{\theta}$ as a *decision rule* which makes use of data to be collected. Bayes risk expresses the expected performance of the decision rule *before* the data is collected, and so is a property of an inference method, rather than a summary of a particular inference. It also depends on the prior distribution. The problem, then, is to determine the estimator $\hat{\theta}$ (or decision rule) which minimizes Bayes risk $B(\pi, \hat{\theta})$. This is known as the *Bayes estimator*. It turns out that a very elegant solution to this problem exists. If L is squared error loss then the Bayes estimator is the mean of the posterior distribution:

$$\hat{\theta}_{MSE} = \int_{\theta \in \Theta} \theta \pi(\theta \mid x) d\theta,$$

which minimizes mean squared error MSE in the sense that it minimizes the mean squared error loss over the prior distribution. Similarly, if L is absolute deviation, then the Bayes estimator is the median of the posterior distribution:

$$\hat{\theta}_{MAD} = median[\pi(\theta \mid x)]$$

which minimizes mean absolute deviation MAD in the sense that it minimizes the mean absolute error loss over the prior distribution.

8.2 Bayesian Inference for the Binomial Distribution

Suppose θ is a probability p in a binomial distribution $bin(n, p)$. If we have no reason to favor one choice of p over the other, we might set $\pi(p)$ to be the uniform distribution on $[0, 1]$. This would correspond to the uniform prior discussed in Section 13.6.1. However, a quite rich theory of Bayesian inference for this problem exists.

8.2.1 The Gamma and Beta Functions

First, the *gamma* function is defined by the definite integral

$$\Gamma(t) = \int_{x=0}^{\infty} x^{t-1} e^{-x} dx, \;\; t > 0.$$

It can be shown that

$$\Gamma(t+1) = t\Gamma(t),$$

and since $\Gamma(1) = 1$ we have

$$\Gamma(n) = (n-1)!$$

for integers $n = 1, 2, 3, \ldots$. The gamma function can therefore be thought of as a generalization of the factorial. In addition, we have

$$\Gamma(1/2) = \sqrt{\pi}.$$

Similarly, the *beta* function is defined by the definite integral

$$B(\alpha, \beta) = \int_{u=0}^{1} u^{\alpha-1}(1-u)^{\beta-1} du, \;\; \alpha, \beta > 0.$$

It can be shown that we always have

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

8.2.2 The Beta Distribution

The beta function is used to normalize the beta distribution $Z \sim beta(\alpha, \beta)$, which has density function

$$f(z \mid \alpha, \beta) = \frac{1}{B(\alpha, \beta)} z^{\alpha-1}(1-z)^{\beta-1}, \;\; z \in [0, 1].$$

The beta distribution has support on the unit unterval $[0, 1]$, and so it is useful for modeling quantities that are interpretable as random probabilities. If $Z \sim beta(1, 1)$ then Z is uniformly distributed on $[0, 1]$, otherwise, the beta family admits a wide variety of shapes. The mean and variance are important to note. We have

$$\begin{aligned} E[Z] &= \frac{1}{B(\alpha, \beta)} \int_{z=0}^{1} z \times z^{\alpha-1}(1-z)^{\beta-1} dz \\ &= \frac{1}{B(\alpha, \beta)} \int_{z=0}^{1} z \times z^{(\alpha+1)-1}(1-z)^{\beta-1} dz \\ &= \frac{B(\alpha+1, \beta)}{B(\alpha, \beta)} \\ &= \frac{\alpha}{\alpha+\beta}, \end{aligned}$$

making use of the equalities noted above. The variance is given by

$$var[Z] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}.$$

It is instructive to reparametrize the beta density, that is, to construct a one-to-one mapping between parameter pairs (α, β) and, say, (ρ, T) such as

$$\begin{aligned} \rho &= \frac{\alpha}{\alpha+\beta}, \\ T &= \alpha+\beta. \end{aligned} \tag{8.1}$$

Under the new parametrization we have

$$\begin{aligned} E[Z] &= \rho, \\ var[Z] &= \frac{\rho(1-\rho)}{T+1}. \end{aligned}$$

This can be compared to the estimator of a binomial proportion $\hat{p} = X/n$, where $X \sim bin(n,p)$, which has mean and variance p and $p(1-p)/n$, respectively (see Chapter 15 of the CSC252 lecture notes). Clearly, $\hat{p}$ resembles $Z \sim beta(\alpha,\beta)$ where, under the parametrization of (8.1), ρ can be equated with p and T can be equated with $n-1$.

8.2.3 Posterior Distribution

Now, suppose we use $beta(\alpha,\beta)$ as the prior density $\pi(p)$ for a binomial parameter p, to construct a posterior distribution for p conditional on observation $X \sim bin(n,p)$. If we wish to use an uninformative prior (Section 13.6.1), we have the uniform prior $p \sim beta(1,1)$. On the other hand, is we believe that p is close to some value p_{prior}, we set $\rho = p_{prior}$ in (8.1). The remaining parameter T reflects the level of certainty we have in this prior assumption, with larger values of T representing greater certainty. In a sense, T can be calibrated by comparison with the sample size n used in the binomial parameter estimate $\hat{p}$ (although $\hat{p}$ is not actually used in this analysis).

Interpreting X as having a binomial distribution conditional on p, we write

$$P(X = x \mid p) = \binom{n}{x} p^x (1-p)^{n-x},$$

leading to posterior distribution

$$\begin{aligned} \pi(p \mid x) &= \frac{P(X=x \mid p)\pi(p)}{\int_{p=0}^{1} P(X=x \mid p)\pi(p)dp} \\ &= \frac{\binom{n}{x} p^x (1-p)^{n-x} \frac{1}{B(\alpha,\beta)} p^{\alpha-1}(1-p)^{\beta-1}}{\int_{p=0}^{1} \binom{n}{x} p^x (1-p)^{n-x} \frac{1}{B(\alpha,\beta)} p^{\alpha-1}(1-p)^{\beta-1} dp}. \end{aligned} \tag{8.2}$$

Although this expression seems complicated, it is actually quite simple, as long as the objective is kept in mind, which is to derive a density of p. We can always express (8.2) in the form

$$\pi(p \mid x) = Kg(p),$$

where K does not depend on p. This is sometimes written as a proportional relationship

$$\pi(p \mid x) \propto g(p).$$

We can then renormalize to get

$$\pi(p \mid x) = \frac{g(p)}{\int_{p=0}^{1} g(p)dp},$$

and the normalization constant may have a convenient form. Clearly, from (8.2) we have

$$\pi(p \mid x) \propto p^{x+\alpha-1}(1-p)^{n-x+\beta-1}.$$

In other words, the posterior density of p given observation $X = x$ is $beta(x + \alpha, n - x + \beta)$. This is an example of a *conjugate prior*, for which the prior and posterior distributions are in the same parametric family. As a technical matter, note that the quantities $\binom{n}{x}$ and $1/B(\alpha, \beta)$ in (8.2) play no role, since they do not depend on p (in fact, they appear in both the numerator and denominator, and so cancel). In particular, we do not need to explicitly evaluate the integral in the denominator.

In the case of the binomial parameter p with $beta(\alpha, \beta)$ prior and observation $X \sim bin(n, p)$, since the posterior density of p given observation $X = x$ is $beta(x+\alpha, n-x+\beta)$, the Bayes estimator with respect to squared error loss is

$$\hat{p}_{MSE} = \frac{x + \alpha}{n + \alpha + \beta}.$$

Chapter 9

Survival Analysis

A *survival time* T is a nonnegative random variable variously interpreted as a *survival time*, *lifetime*, *waiting time*, *time to event*, and so on. The geometric and exponential RVs (Sections 4.10.4, 4.11.2 of CSC262 lecture notes) are two examples of survival times, so a survival time may be discrete (for example, number of integer days until event) or continuous. Recall the memoryless property which characterizes the geometric and exponential distributions, that the average waiting time remaining, after having already waited a time t, is the same as the original average waiting time.

The key to understanding this idea is to consider a *failure rate*. Suppose a survival time T is discrete, representing the number of days until failure of a component (a light bulb, for example). Let q_i be the failure rate on day i. This means that if the component has survived until day i ($T \geq i$) we toss a coin (independently of all previous coin tosses). If we get a 'head' (for our particular coins, this has probability q_i) the component 'fails' and the survival time is $T = i$, terminating the process. First, note that the failure rates q_i are not a probability mass function for T. They are actually the conditional probabilities

$$q_i = P(T = i \mid T \geq i), \;\; i = 1, 2, \ldots.$$

Second, these rates may increase or decrease in time, and what defines a memoryless distribution is precisely the assumption that the failure rates remain constant. This defines the geometric distribution.

We do not, on the other hand, expect the lifetime of, say, a car to be memoryless. We expect that the probability that a 10-year old car survives one more year is smaller than that for a 5-year old car, and smaller still than that for a new car. In other words, the failure rates q_i increase in i. Such a survival time is called *new better than used (NBU)*.

A survival time may also be *new worse than used (NWU)*, in which case the failure rates are decreasing. The survival time for young members of a species in an environment with high infant mortality will typically be NWU. This is because the period immediately after birth is very high in mortality risk, meaning that the failure rate is correspondingly high. However, if the infant survives this high risk period, the failure rate will decrease, resulting in a NWU survival time. Of course, if the infant survives into adulthood, the failure rate will begin to increase. A natural source of NWU survival times would be survival in competitive environments.

Example 9.1. A random variable W has a *Weibull distribution* if there are two parameters $k > 0$ and $\lambda > 0$ such that

$$X = (\lambda W)^k \tag{9.1}$$

has an $exp(1)$ distribution (exponential distrubution with $\lambda = 1$). This will be denoted $W \sim weibull(k, \lambda)$. This distribution is commonly used to model survival times. By convention, k is the *shape parameter* and θ is the *rate parameter*. Note that in some conventions λ is replaced by, say, $1/\tau$, in which case τ is referred to as a *scale parameter* (be careful, since λ may be used as a scale parameter). Both definitions are equivalent, once the transformation is understood. We use the rate parameter in order to emphasize the relationship with the exponential distribution.

Suppose $W \sim weibull(k, \lambda)$. The CDF of $X \sim exp(1)$ is $F_X(x) = 1 - \exp(-x)$ for $x \geq 0$, so

$$F_W(w) = P(W \leq w) = P\left(\lambda^{-1} X^{1/k} \leq w\right) = P\left(X \leq (\lambda w)^k\right) = 1 - \exp\left(-(\lambda w)^k\right), \;\; w \geq 0,$$

and $F_W(w) = 0$ for $w < 0$. To evaluate the density function, take the deriviate of the cumulative distribution function (CDF), giving

$$f_W(w) = \frac{d}{dw}\left\{1 - \exp\left(-(\lambda w)^k\right)\right\} = k\lambda^k w^{k-1} \exp\left(-(\lambda w)^k\right), \;\; w \geq 0,$$

and $f_W(w) = 0$ for $w < 0$.

For some positive d, define the function

$$h(x; d, k, \lambda) = P(W \geq x + d \mid W \geq x) = \frac{P(W \geq x + d)}{P(W \geq x)}.$$

This is interpretable as the probability that a system with a lifetime of W survives an additional d time units, given that it has survived x time units. We may write an `R` function that accepts input (x, d, k, λ) and returns $h(x; d, k, \lambda)$. Note that the `R` function `dweibull` uses the scale parameter, not the rate parameter, so we need to transform accordingly.

```
f0 = function(x,d,ishape,irate) {
  pweibull(x+d,shape=ishape,scale=1/irate,lower.tail=F)
               /pweibull(x,shape=ishape,scale=1/irate,lower.tail=F)
}
```

Consider 3 Weibull distributions:

$$\begin{aligned} W_1 &\sim weibull(k = 1/2, \lambda = 1/5), \\ W_2 &\sim weibull(k = 1, \lambda = 1/10), \\ W_3 &\sim weibull(k = 3/2, \lambda = \sqrt{\pi}/20). \end{aligned}$$

The following `R` code draws the required plot in Figure 9.1.

```
ex1 = expression(italic(x))
ex2 = expression(paste(italic(h),'(',italic(x),';',italic(d),',
      ',italic(k),',',lambda,')',sep=''))
ex3 = expression(paste(italic(d), ' = 1, ',italic(k),' = 1/2,
     ',lambda,' = 1/5',sep=''))
ex4 = expression(paste(italic(d), ' = 1, ',italic(k),' = 1,
     ',lambda,' = 1/10',sep=''))
ex5 = expression(paste(italic(d), ' = 1, ',italic(k),' = 3/2,
```

```
     ',lambda,' = ',sqrt(pi),'/20',sep=''))

xgr = seq(0,100,by = 1)
y = cbind(f0(xgr,1,0.5,1/5), f0(xgr,1,1.0,1/10), f0(xgr,1,2.0,sqrt(pi)/20))
par(mar=c(5,5,5,5))
matplot(xgr,y, col=c(2,3,4),lty=1,type='l',xlab=ex1,ylab=ex2)
legend('bottomleft',legend = c(ex3,ex4,ex5),col=c(2,3,4),lty=1,bty='n')
```

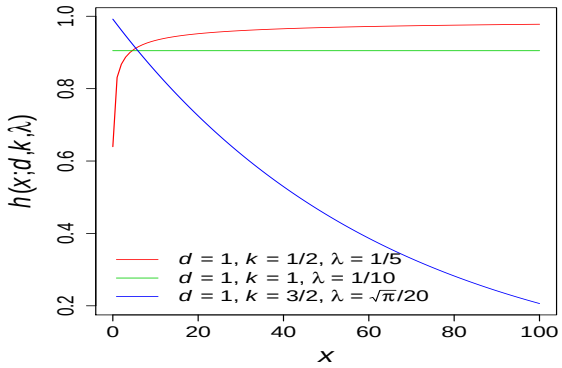

Figure 9.1: Plot for Example 9.1

The probability of surving an additional time unit given survival up to time x increases for W_1, remains constant for W_2, and decreases for W_3. This means W_1, W_2, W_3 are NWU, memoryless and NBU, respectively. See Figure 9.1. □

The quantity $h(x; d, k, \lambda)$ of Example 9.1 is interpretable as a probability of surviving an addition d time units. If we divide by d, then the quantity $h(x; d, k, \lambda)/d$ can be interpreted as a *failure rate* or *hazard rate*. In fact, we can show that if we allow d to approach zero, the limit is precisely the *hazard function* of a survival time X:

$$h(x) = \frac{f(x)}{S(x)},$$

where $S(x)$ is the *survival function*

$$S(x) = 1 - F(x)$$

and $f(x), F(x)$ are the density function and CDF of X. Then $h(x)$ is interpreted as the failure rate at time x. The *cumulative hazard functio*n is simply the integral

$$H(x) = \int_{y=0}^{x} h(y)dy.$$

and it can be shown that

$$H(x) = -\log(S(x)).$$

9.1 Estimation of the Survival Function

The survival function is of great importance in survival analysis. For example, a cancer prognosis is usually given in terms of $S(x)$. The statement '5 year survival is 30%' means exactly $S(5) = 0.3$. Since $S(x)$ is simply the complement of the CDF $F(x)$, if we are given a sample of survival times $x_1, \ldots, x_n$, we can first estimate F with the *empirical distribution function*

$$\hat{F}(x) = \frac{\text{number of } x_i \leq x}{n},$$

then the estimated survical function is

$$\hat{S}(x) = 1 - \hat{F}(x). \tag{9.2}$$

9.1.1 Censoring

Unfortunately, there is a feature common to samples of survival data that prohibits the use of (9.2). Suppose that we are studying the survival times of cancer patients (from time of diagnosis until death by cancer, for example). The survival time t_i recorded for patient i will, in practice, either be the time from diagnosis to death by cancer, or it will be the time that the patient was observed without having died from cancer. Presumably, a patient can be followed up only within the lifetime of the study, or it may be that the patient died of other causes, or left the study for any number of reasons. In this case we say that the observation t_i is *censored*. It is a partial observation of the survival time, in the sense that we can only say that the cancer survival time is $\geq t_i$. But, this is still useful information, and so should be incorprated into the analysis. Note there are are other forms of censoring. The one we have described is known as *right censoring*.

The symbol '+' is used to denote censoring. If we have data (in months)

$$10.3, 11.2+, 13.6, 15.2$$

this might mean, for example, that three patients died from cancer 10.3, 13.6 and 15.2 months after diagnosis, and one patient was observed for 11.2 months after diagnosis without having died from cancer.

9.1.2 Kaplan - Meier Estimate of the Survival Function

Censored data cannot be used to construct the survival function estimate (9.2). Instead we may use the *Kaplan - Meier estimate*. Suppose we are given survival times

$$0 = t_0 < t_1 < t_2 < \ldots < t_{m-1} < t_m. \tag{9.3}$$

Define intervals

$$I_i = [t_i, t_{i+1}).$$

Next, suppose p_i is the probability of surviving interval I_i, given survival up to time t_i. Then

$$S(t_i) \approx \prod_{j=0}^{i-1} p_j.$$

To estimate p_i, let $r(t_i)$ be the number at risk (still alive) just before time t_i, and let d_i be the number of deaths. Then we estimate

$$p_i \approx \hat{p}_i = \frac{r(t_i) - d_i}{r(t_i)}.$$

The Kaplan - Meier estimate of the survival function is then

$$\hat{S}(t) = \prod_{t_i<t} \hat{p}_i.$$

The estimator has a natural tabular representation. Suppose we are given a sample of survival times $T_1, \ldots, T_n$. Values may be represented more than once, and some observations are censored. Suppose the represented values are sorted as in (9.3). Survival time 0 is included as t_0 whether or not it appears in the sample. Thus, m need not equal n. We can then construct table

i	t_i	d_i	$r(t_i)$	$\hat{p}_i$
0	$t_0 = 0$	d_0	$r(t_0) = n$	$(r(t_0) - d_0)/r(t_0)$
1	t_1	d_1	$r(t_1)$	$(r(t_1) - d_1)/r(t_1)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m	t_m	d_m	$r(t_m)$	$(r(t_m) - d_m)/r(t_m)$

Example 9.2. For example, if we have times $23.5, 34.0+, 34.0, 39.1+, 43.7$, this yields table:

i	t_i	d_i	$r(t_i)$	$\hat{p}_i$
0	0	0	5	(5-0)/5 = 1
1	23.5	1	5	(5-1)/5 = 4/5
2	34.0	1	4	(4-1)/4 = 3/4
2	39.1	0	2	(2-0)/2 = 1
3	43.7	1	1	(1-1)/1 = 0

The resulting Kaplan-Meier estimate is shown in Figure 9.2. Note that censored observations are usually indicated in a plot by the symbol '+'.

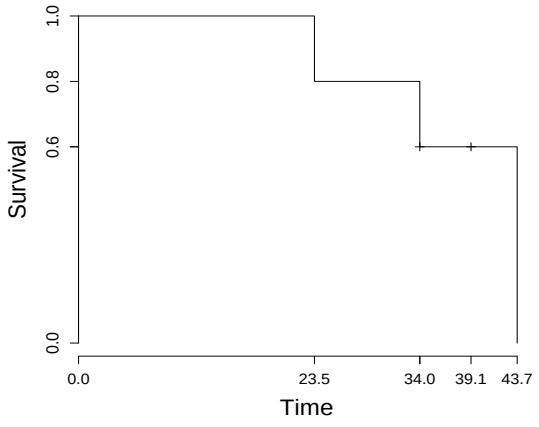

Figure 9.2: Plot for Example 9.2

□

9.1.3 Cox Proportional Hazards Regression

There are various ways to incorporate linear models into survival analysis. One of the most widely used is the *Cox proportional hazards regression model*. We have the same linear prediction term used in Gaussian linear models and logistic regression:

$$\eta = \beta_1 x_1 + \ldots + \beta_p x_p.$$

There is a *baseline hazard function* $h_0(x)$. The crucial assumption is that for any set of predictions, there is an associated survival time with a hazard function $h(x) \propto h_0(x)$, the exact relationship being

$$h(x) = h_0(x)e^{\eta}.$$

The data consists of observed survival times (possibly censored) associated with a set of predictors. Note that the intercept term β_0 is not needed. The important quantity is the *hazard ratio*

$$HR = e^{\eta - \eta'},$$

where η and η' are the linear prediction terms for two sets of predictors. This plays a similar role to the odds ratio in logistic regression (Section 13.9.1).

Part II

Computational Methods

Chapter 10

Simulation Methods

Much of the theory we have seen is based on approximations to the normal distribution. We either assume that the measurements we collect are normally distributed, or that the sample size is large enough for the Central Limit Theorem to apply to the test statistic. This applies also to test statistics with a χ^2 distribution used for categorical data, with the additional assumption that each category count is large enough.

There is very good reason to study normal-based theory, since much of it is provably optimal when the assumptions are satisfied, as they often are. However, these assumptions will often prove problematic, and even when they may hold, it might not be practical to verify them, for example, when a procedure is to be repeatedly applied to a large number of cases. This is generally the case in the analysis of, for example, gene expression data.

We then briefly consider two forms of simulation methods which do not rely on distributional assumptions, and which are generally applicable.

10.1 Permutation Test

A *permutation test* is a hypothesis test in which a null distribution is created by a random permutation of the data. Consider the following paired data,

```
X = 16.1 31.5 21.5 22.4 20.5 28.4 30.3 25.6 32.7 29.2 34.7
Y = 4.41 6.81 5.26 5.99 5.92 6.14 6.84 5.87 7.03 6.89 7.87
```

for which the Pearson correlation coefficient is $r_{obs} = 0.939$. Suppose we randomly permute one of the variables (say, Y), then recalculate r. We can generate a random permutation with the `sample()` function:

```
> sample(11)
 [1]  8  9  4  5  6 11  3  2  1  7 10
>
```

We can then permute Y, then recalculate r:

```
> Yrandom = Y[sample(11)]
> X
 [1] 16.1 31.5 21.5 22.4 20.5 28.4 30.3 25.6 32.7 29.2 34.7
```

```
> Yrandom
 [1] 4.41 6.81 6.84 7.03 6.14 6.89 5.26 7.87 5.87 5.99 5.92
> cor(X,Y)
[1] 0.9388037
> cor(X,Yrandom)
[1] 0.1196821
>
```

The correlation of the permuted data, $r^* = 0.1196821$, is much smaller than the original $r_{obs} = 0.939$. This number is quite relevant, however. Under the null hypothesis $H_o : \rho = 0$, there is no association between the paired variables X and Y. Therefore, if H_o is true, the observed sample correlation r_{obs} should be comparable to a correlation coefficient r^* produced by randomly permuting the data. This gives directly a test procedure that does not require any distribution assumptions. We can estimate the *null distribution* of r^* by repeatedly permuting the data, and then compared r_{obs} to this distribution, either by comparing it to a critical value of the null distribution, or by estimating the appropriate tail probability to obtain a p-value.

We first simulate r^* $N = 50,000$ times, and display the distribution in a histogram (Figure 10.1).

```
> r.perm = rep(NA,50000)
> for (i in 1:50000) {r.perm[i] = cor(X,Y[sample(11)])}
>
> hist(r.perm, nclass=25)
> lines(rep(0.735,2), c(0,5000), col=4)
> lines(rep(-0.735,2), c(0,5000), col=4)
> text(-0.735,5500,"r = -0.735")
> text(0.735,5500,"r = 0.735")
```

We can show that the critical value $r_{\alpha/2}$ for a two-sided test against $H_o : \rho = 0$ ($\alpha = 0.01$, $n = 11$) was $r_{\alpha/2} = 0.735$, that is we reject H_o if $|r_{obs}| \geq 0.735$. This critical value is shown in Figure 10.1. This means that under the null distribution, the correlation coefficient satisfies:

$$P(|r| \geq 0.735 \mid \rho = 0) = 0.01.$$

We can estimate the same probability for r^* from the simulated null distribution:

```
> mean(abs(r.perm) >= 0.735)
[1] 0.00862
```

so that

$$P(|r^*| \geq 0.735 \mid \rho = 0) \approx 0.0082,$$

which is close to $\alpha = 0.01$. In fact, the level 95% margin of error of an estimate of a proportion $p = 0.01$ with $n = 50,000$ is

$$ME = 1.96\sqrt{\frac{0.01 \times 0.99}{50000}} = 0.00087.$$

Judging from the margin of error, the tail probabilities for 0.735 is close to, but slightly less than, $\alpha = 0.01$. We can obtain a critical value for r_{obs} based on the distribution of r^* using the *quantile*() function:

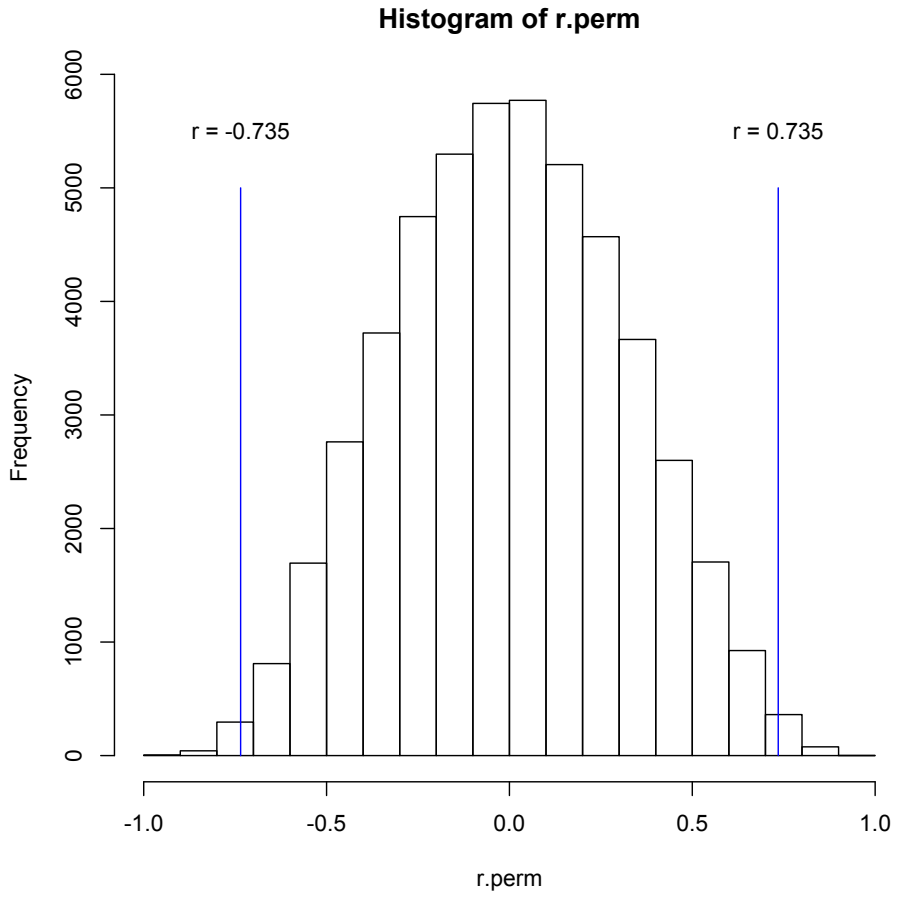

Figure 10.1: Histogram of 50,000 replications of r^*. The critical values $r_{\alpha/2}$, $-r_{\alpha/2}$ for a two-sided test against $H_o : \rho = 0$ ($\alpha = 0.01$, $n = 11$) are superimposed.

```
> quantile(abs(r.perm),0.99)
      99%
0.7257387
> quantile(r.perm,0.995)
   99.5%
0.734525
> quantile(r.perm,0.005)
      0.5%
-0.7181912
>
```

Note that we can obtain the $\alpha/2 = 0.005$ critical value for r^* ($r^*_{0.005} = 0.73425$), the lower tail $1 - \alpha/2 = 0.995$ critical value ($r^*_{1-0.005} = -0.7181912$), or the $\alpha = 0.01$ critical value for $|r^*|$ ($|r^*|_{0.01} = 0.7257387$). If the null distribution is symmetric, which we would expect in this case,

the critical value $|r^*|_{0.01}$ should be used, since we would expect

$$|r^*|_\alpha \approx r^*_{\alpha/2} \approx -r^*_{1-\alpha/2}.$$

The p-value can be obtained by estimating the relevant tail probability of r_{obs}, in this case

$$P(r^* \geq r_{obs}), \;\; P(r^* \leq r_{obs}), \quad \text{or } P(|r^*| \geq |r_{obs}|)$$

for an upper-tailed, lower-tailed or two sided test, respectively. However, it is usually the practice to add the observed statistics r_{obs} with the simulated values of r^* for this purpose, giving, for example:

$$P(|r^*| \geq |r_{obs}|) \approx \frac{\#\{|r^*| \geq |r_{obs}|\} + 1}{N + 1}. \tag{10.1}$$

This avoids p-values equal to zero, making the procedure somewhat conservative, although less so with increasing N. To assign a p-value to $r_{obs} = 0.939$, we can determine the numerator of (10.1) with the following command:

```
> sum(abs(r.perm) >= 0.939)
[1] 0
>
```

that is, no simulated value of r^* exceeds 0.939 in magnitude. This gives p-value

$$P \approx 1/50001 = 1.99996 \times 10^{-5}.$$

10.2 The Bootstrap Procedure

Suppose we are given a sample of size $n = 10$:

```
X = 36.1 16.1 16.7 32.7 33.9 21.8 15.5 26.0 37.8 18.6
```

A 95% confidence interval for the mean is given by

$$\bar{X} \pm t_{9,0.025} S/\sqrt{n} = 25.2 \pm 6.37 = (19.15, 31.89).$$

Remember that a confidence interval is a statement about a statistical method as well as a specific data set. If we could observe repeated samples collected under identical conditions, obtaining repeated observations of $\bar{X}$, we could observe the distribution of $\bar{X}$ directly, and form inference statements accordingly, without the need to specify a distribution.

The *bootstrap procedure* is a method of simulating such samples, thus obtaining an estimated *sampling distribution* of, for example, $\bar{X}$, or any other statistic of interest. This is done by the simple device of sampling, *with replacement*, from the original sample (of size n), a new sample of the same size n.

This can be done in R by the `sample()` function in the following way:

```
> n = 10
> sample(1:n, n, replace = TRUE)
 [1] 1 6 4 2 3 5 5 8 8 3
>
```

(the command `sample.int(n, n, replace = TRUE)` will do the same thing). A *bootstrap sample* is then obtained by replacing the indices in the original sample:

```
> Xboot = X[sample(1:n, n, replace = TRUE)]
> X
 [1] 36.1 16.1 16.7 32.7 33.9 21.8 15.5 26.0 37.8 18.6
> Xboot
 [1] 33.9 21.8 16.1 37.8 36.1 15.5 16.7 32.7 21.8 16.1
> mean(X)
[1] 25.52
> mean(Xboot)
[1] 24.85
>
```

The bootstrap sample contains repeats, but we can still calculate most statistics for it. In this case, we get a new sample mean $\bar{X}_{boot} = 24.85$ close to, but not exactly equal to, to original observed sample mean $\bar{X}_{obs} = 25.52$. As for the permutation procedure, we may then obtain a simulated sample, shown as a histrogram in Figure 10.2.

```
> xbar.boot = rep(NA,50000)
> for (i in 1:50000)
             {xbar.boot[i] = mean(X[sample(1:n, n, replace = TRUE)])}
> hist(xbar.boot, nclass=25)
>
```

To obtain a 95% confidence interval, we need only obtain the 0.025 and 0.975 quantiles from the bootstrap sample,

```
> quantile(xbar.boot, c(0.025, 0.975))
 2.5% 97.5%
20.40 30.81
>
```

yielding an estimated 95% confidence interval of $(20.40, 30.81)$, which is quite close to the confidence interval $(19.15, 31.89)$ obtained using the t-distribution above.

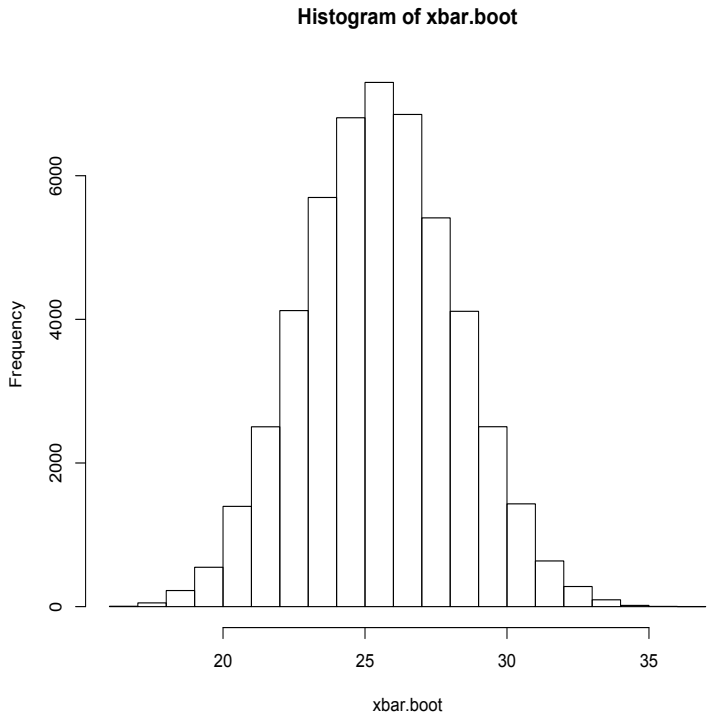

Figure 10.2: Histogram of 50,000 replications of $\bar{X}_{boot}$.

Chapter 11

MCMC Simulation and Bayesian Inference

11.1 Markov Chains

A *stochastic process* may be defined as a (possibly uncountable) indexed collection of random variables $\{X_t\}$, $t \in \mathcal{T}$. The index t usually represents time, although stochastic processes may also be used to describe random processes defined on some space.

Most stochastic processes are either *discrete time*, and take the from of a sequence $X_1, X_2, \ldots$, or continuous time, and may be represented as a process $X[t]$ on a subset $t \in [0, \infty)$, with $X[t]$ being the value of the process at time t.

The *Markov chain* is a discrete time stochastic process. The defining property is the *memoryless property* or *Markovian property*, essentially, that the distribution of future states depends on the current state, but not on previous states.

Definition 11.1. Suppose we are given a discrete time stochastic process $X_n \in \mathcal{X}$, $i = 0, 1, 2, \ldots$, which assumes values in a discrete *state space* $\mathcal{X}$. Without loss of generality we have either a finite state space $\mathcal{X} = \{0, 1, \ldots, n\}$ or countable state space $\mathcal{X} = \{0, 1, \ldots\}$. Then X_i is a *Markov chain* if the following *memoryless property* holds:

$$P(X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, \ldots, X_1 = i_1, X_0 = i_0) = P(X_{n+1} = j \mid X_n = i) = P_{ij}.$$

The quantity P_{ij} is called the *transition probability* from state i to state j. We also have *transition probability matrix* (or *transition matrix*)

$$P = \begin{bmatrix} P_{00} & P_{01} & P_{02} & \cdots \\ P_{10} & P_{11} & P_{12} & \cdots \\ \vdots & \vdots & \vdots & \\ P_{i0} & P_{i1} & P_{i2} & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

□

Row i of transition matrix P is equivalent to the conditional probability

$$P(X_{n+1} = j \mid X_n = i) = P_{ij}, \;\; j \in \mathcal{X}.$$

Note also that P will be a matrix of infinite dimension when $\mathcal{X}$ is countable. We also have no difficulty conceiving of P as 'doubly infinite' when the state space is the set of positive and negative integers $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$, which requires no important change of Definition 11.1.

Example 11.1. We start with an example of a two-state Markov chain, which, despite it's simplicity, demonstrates a number of important features of Markov chains. Formally, we have state space $\mathcal{X} = \{0, 1\}$. However, we lose nothing by replacing the notation of Definition 11.1 with something more intuitive.

For example, the time index $i = 0, 1, 2, \ldots$ may represent a sequence of days, and we may wish to define a simple infection model, in which state $i = 0$ represents a *healthy state* H and $i = 1$ represents a *sick state* I (due to, say, an infection). The transition matrix is therefore the 2×2 matrix

$$P = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}.$$

However, the true degrees of freedom of P is 2, since each row is constrained, as a probability distribution, to sum to 1 (such a matrix is known as a *stochastic matrix*). We can therefore write, without loss of generality,

$$P = \begin{bmatrix} 1 - \alpha & \alpha \\ \beta & 1 - \beta \end{bmatrix}. \tag{11.1}$$

for two numbers $\alpha, \beta \in [0, 1]$. This means that if a subject is healthy on day i, he/she is sick on day $i + 1$ with probability α, and if the subject is sick on day i, he/she is sick on healthy in day $i + 1$ with probability β. The state transition diagram for infection model is shown in Figure 11.1.

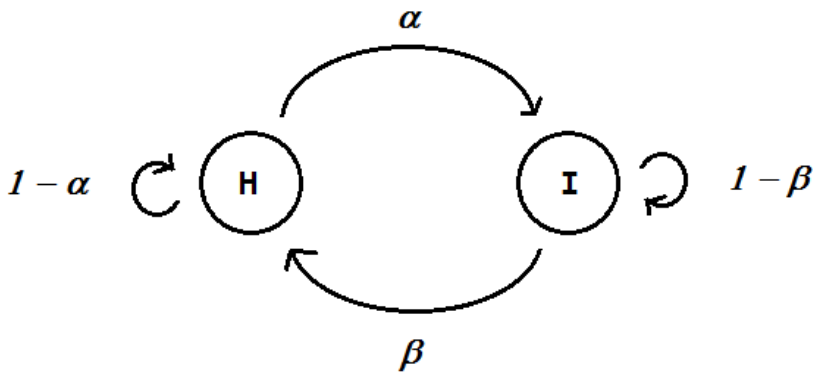

Figure 11.1: State transition diagram for infection model of Example 11.1.

Is this a reasonable model? First, we note that when the subjects enters state H, he/she remains there for a geometrically distributed waiting time, with mean α^{-1}. If we suppose that acquiring an

infection is a consequence of a chance exposure, which happens on any given day with probability α, then the memoryless 'coin toss' model for the waiting time until infection would be reasonable.

On the other hand, the infection lifetime also follows a geometric distribution, but with mean β^{-1}. Presumably, clinical experience would guide the choice of β, setting

$$\beta^{-1} = E[\text{ infection lifetime }].$$

However, whether the geometric distribution adequately models an infection lifetime would be an important question to resolve. □

The transition probability P_{ij} may be more formally referred to as the *one-step transition probability*, since it describes transition following a single time step. We may also describe the k-step transition probability

$$P(X_{n+k} = j \mid X_n = i) = P_{ij}^k, \tag{11.2}$$

noting that this probability does not depend on n. We will demonstrate this computation for $k = 2$. In Equation (11.2) set

$$E = \{X_{n+2} = j\}, \;\; B = \{X_n = i\},$$

and we may form partition

$$A_k = \{X_n = k\} \text{ for all } k \in \mathcal{X}.$$

This gives

$$P_{ij}^2 = P(X_{n+2} = j \mid X_n = i) = \sum_{k \in \mathcal{X}} P(X_{n+2} = j \mid X_{n+1} = k, X_n = i) P(X_{n+1} = k \mid X_n = i). \tag{11.3}$$

Then consider each term in the summation. Recall by the Markovian property of Definition 11.1 that the distribution of X_{n+2} given history $X_{n+1}, X_n, \ldots, X_1, X_0$ depends only on the most recent state X_{n+1}. We therefore have

$$P(X_{n+2} = j \mid X_{n+1} = k, X_n = i) = P(X_{n+2} = j \mid X_{n+1} = k) = P_{kj}. \tag{11.4}$$

The remaining quantity is simply the one step transition probability

$$P(X_{n+1} = k \mid X_n = i) = P_{ik}. \tag{11.5}$$

Substituting (11.4) and (11.5) into (11.3) yields

$$P_{ij}^2 = \sum_{k \in \mathcal{X}} P_{ik} P_{kj}. \tag{11.6}$$

This is a particularly important relationship, since we can recognize (11.6) as the result of matrix multiplication. We summarize this in the following definition.

Definition 11.2. The *k-step transition probability* from state i to j is the probability that a Markov chain in state i occupies state j after exactly k transitions. Formally,

$$P(X_{n+k} = j \mid X_n = i) = P_{ij}^k,$$

The k-step transition probability matrix P^k has value P_{ij}^k at (row,column) (i, j), and can be calculated by iteratively multiplying P k times:

$$P^k = [P]^k$$

□

11.1.1 Balance Equations and Steady States

There is often interest in the long run behavior of a Markov chain. In Example 11.1 the process fluctuates between *Healthy* and *Infected* states indefinitely, and we may be interested in knowing the long run proportion of time spent in each state.

Suppose we have counting process

$$N_j(k) = \text{ The number of transitions into state } j \text{ after the } k\text{th transition.}$$

A long run frequency (formally, a *steady state frequency*) would then be defined by

$$\pi_j = \lim_{k\to\infty} \frac{N_j(k)}{k}. \tag{11.7}$$

A number of mathematical questions lurk here. Does the limit always exist? If so, under what conditions is π_j zero or positive? Last, and far from least, does this quantity depend on the initial state? To formally resolve these questions requires some amount of mathematical theory, even when dealing with relatively simple models. While this would be beyond the scope of this course, we can give some insight into the issues.

Deducing π_j for general models requires acknowledgement of an apparently obvious fact, that is,

$$\text{number of times a Markov chain enters a state} = \text{number of times a Markov chain exits a state,} \tag{11.8}$$

(plus or minus one). The value of this statement becomes more apparent if we think in terms of *rates*. Clearly, π_j can be interpreted as the *occupancy rate* of state j, but also as its *entrance rate* and the *exit rate*. Next, we may consider the rate at which the Markov chain transitions from states i to j. There are two components to this. First, to transition from i to j, the Markov chain must first enter i. This occurs at rate π_i. Second, given that the Markov chain is in i, it transitions from i to j with probability P_{ij}. The rate of transition from i to j is therefore $\pi_i P_{ij}$.

We are now in a position to use (11.8). We recognize the exit rate for state j as π_j. The entrance rate, on the other hand, can be given as the sum of all other transition rates *into* state j, that is , $\pi_i P_{ij}$ for $i \in \mathcal{X}$ (this includes $i = j$, when $P_{jj} > 0$). Equation (11.8) then yields the *balance equation*

$$\pi_j = \sum_{i\in\mathcal{X}} \pi_i P_{ij}. \tag{11.9}$$

Example 11.2. Consider the two-state Markov chain of Example 11.1. We write a balance equation for each state, yielding

$$\begin{aligned} \pi_0 &= \pi_0 P_{00} + \pi_1 P_{10} \\ \pi_1 &= \pi_0 P_{01} + \pi_1 P_{11}, \end{aligned}$$

which, after substituting transition probabilities (11.1) gives

$$\begin{aligned} \pi_0 &= \pi_0(1-\alpha) + \pi_1\beta \\ \pi_1 &= \pi_0\alpha + \pi_1(1-\beta). \end{aligned}$$

Rewriting the first equation yields

$$\frac{\pi_0}{\pi_1} = \frac{\beta}{\alpha}. \tag{11.10}$$

Since we must have $\sum_i \pi_i = 1$, only one balance equation is actually needed to solve for (π_0, π_1), and we obtain

$$\pi_0 = \frac{\beta}{\alpha+\beta}, \;\; \pi_1 = \frac{\alpha}{\alpha+\beta}.$$

That the frequencies π_0, π_1 should posses ratio β/α is to be expected. The time spent in each state prior to transition is geometrically distributed with means $1/\alpha$ and $1/\beta$ respectively. The ratio π_0/π_1 should then be the ratio of the means, which is confirmed by (11.10). □

11.2 The Hastings-Metropolis algorithm

In Bayesian inference, we have a posterior density of the form

$$\pi(\theta \mid x) = \frac{f(x \mid \theta)\pi(\theta)}{f(x)} = \frac{f(x \mid \theta)\pi(\theta)}{\int_\Theta f(x \mid \theta)\pi(\theta)d\theta},$$

or, similarly when model space Θ is discrete,

$$\pi(\theta \mid x) = \frac{f(x \mid \theta)\pi(\theta)}{f(x)} = \frac{f(x \mid \theta)\pi(\theta)}{\sum_{\theta\in\Theta} f(x \mid \theta)\pi(\theta)}.$$

It is often not possible to evaluate the normalization constant (which is $f(x)$ in this example). In this case, we may use a Markov chain Monte Carlo method to simulate a sample from a density or probability mass function, say $p(y)$, on state space $\mathcal{S}_y$, which is known only up to a normalizing constant. This means we can write

$$p(y) = Kg(y),$$

where $g(y)$ is known, K does not depend on y but K is otherwise unknown. We can, for example, write a Bayesian posterior density

$$\pi(\theta \mid x) = Kg(\theta)$$

where, for fixed x, $g(\theta) = f(x \mid \theta)\pi(\theta)$ and $K = 1/f(x)$.

The *Hastings-Metropolis algorithm* is an MCMC method which simulates a Markov chain on a state space $\mathcal{S}_y$ which has steady state distribution $p(y)$. It depends on $p(y)$ only through ratios of the form $p(y')/p(y)$. Therefore, if $p(y) = Kg(y)$, where K does not depend on y, the algorithm only needs ratios

$$\frac{p(y')}{p(y)} = \frac{Kg(y')}{Kg(y)} = \frac{g(y')}{g(y)},$$

that is, all we need is a function $g(y)$ which is *proportional* to $p(y)$.

The algorithm takes the following steps:

- First define $Q(y_2 \mid y_1)$, a *proposal* Markov chain on $\mathcal{S}_y$. This MC should be *irreducible*, which can in practice be difficult to prove (a MC is *irreducible* if any state may be reached from any other state).

- Construct a Markov chain $y_1, y_2, \ldots$ according to the following algorithm:
  1. Given current state y_n, select *proposal state* y' according to probability distribution $Q(y' \mid y_n)$.
  2. With probability α set $y_{n+1} = y'$, and with probability $1 - \alpha$ set $y_{n+1} = y_n$, where
$$\begin{aligned} r &= \frac{p(y')Q(y_n \mid y')}{p(y_n)Q(y' \mid y_n)} \\ \alpha &= \min(r, 1). \end{aligned}$$
- The Markov chain is irreducible, with steady-state distribution $p(y)$.

There are alternative formulations. See Hastings (1970) "Monte Carlo sampling methods using Markov chains and their applications", *Biometrika*.

11.3 Simulated annealing

Suppose the objectve is to maximize a function $f(y)$ on a state space $\mathcal{S}_y$ (or minimize $-f(y)$). This can be done using *simulated annealing*. This is an MCMC algorithm similar to the Hastings-Metropolis algorithm, in that it simulates a stochastic process $y_1, y_2, \ldots$ using a similar proposal acceptance mechanism. The difference is that, under known conditions, the process converges to y^*, where $\max_y f(y) = f(y^*)$. In particular, if $\mathcal{S}_y$ is discrete, then

$$\lim_{n\to\infty} P(y_n = y^*) = 1,$$

assuming the maximum is unique. Under quite general conditions, we have for any positive constant $\delta > 0$

$$\lim_{n\to\infty} P(f(y_n) > f(y^*) - \delta) = 1.$$

Many optimization algorithms guarantee convergence to a local maximum, but simulated annealing is one of the few optimization algorithms which is able to guarantee convergence to the global maximum
The algorithm takes the following steps:

- Construct a proposal Markov chain $Q(y' \mid y_n)$ as in the Hastings-Metropolis algorithm.
- Define a decreasing *temperature* sequence $t_1, t_2, \ldots$ (the *cooling schedule*).
- Construct a process $y_1, y_2, \ldots$ according to the following algorithm:
  1. Given current state y_n, select *proposal state* y' according to probability distribution $Q(y' \mid y_n)$.
  2. With probability α set $y_{n+1} = y'$, and with probability $1 - \alpha$ set $y_{n+1} = y_n$, where
$$\alpha = \begin{cases} 1 & \text{if } f(y') \geq f(y_n) \\ \exp\left(\frac{f(y') - f(y_n)}{t_n}\right) & \text{if } f(y') < f(y_n) \end{cases}$$
- The process $y_1, y_2, \ldots$ converges to the maximum if $f(y)$ in the sense given above.

Note that the process $y_1, y_2, \ldots$ is Markovian, or memoryless, but is not a time-homogenous Markov chain, since the transition probabilities depend on time index n through the cooling schedule.

We next consider the question of the the cooling schedule (a good review can be found in Nourani and Andresen (1998) "A comparison of simulated annealing cooling strategies." *Journal of Physics A: Mathematical and General*). For a cooling schedule of the form

$$t_n = \frac{c}{\log(n+d)}, \tag{11.11}$$

the simulated annealing has been proven to converge to the optimal solution in the sense defined above, provided c is large enough. Unfortunately the minimum value of c for which convergence can be guaranteed depends on the problem, and in general, the extremely slow convergence rate makes this cooling schedule impractical. Furthermore, because the algorithm is stochastic, it will not generally be possible to define a stopping rule, that is, a rule which can be used to decide when the optimal solution has been reached (assuming the largest value $f(y^*)$ is not known).

Despite these qualifications, simulated annealing is useful for optimization problems not possessing regularity conditions for which more specialized algorithms would be available. It is also widely applicable, and relatively easy to implement. In practice the cooling schedule (11.11) is not used. Commonly used choices include the exponential schedule

$$t_n = t_0 \rho^n$$

and the linear schedule

$$t_n = t_0 - \beta n,$$

although neither results in a provably convergent algorithm.

Part III

Supervised and Unsupervised Learning

Chapter 12

Machine Learning and Statistical Learning - General Concepts

Machine learning describes computer applications in pattern recognition, computational learning or artificial intelligence. Examples include the detection of purposeful motion in a dynamic pixel field, or the development of software capable of identifying handwritten letters. Statistical learning is the application of statistical methodology to these problems, focusing on the exploitation of data sets.

Most statistical learning problems share a common structure. We are given, at the very least, a set of n *observations* $\dot{x}_1, \ldots, \dot{x}_n$. For our purposes, we can take $\dot{x}_i$ to be a vector of length p:

$$\dot{x}_i = (x_{i1}, \ldots, x_{ip}).$$

If we combine the observations into rows, we get an $n \times p$ matrix

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ip} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_i \\ \vdots \\ \dot{x}_n \end{bmatrix}.$$

It is important to note that $\mathbf{X}$ may also be decomposed by column. We may also define vector $\mathbf{x}_j = (x_{1j}, \ldots, x_{nj})$, so that

$$\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_p].$$

Formally, we are interpreting $\dot{x}_i$ as a $1 \times p$ row vector, and $\mathbf{x}_j$ as a $n \times 1$ column vector, that is

$$\dot{x}_i = [x_{i1} \cdots x_{ip}] \text{ and } \mathbf{x}_j = \begin{bmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{bmatrix}.$$

The symbol $\mathbf{X}$ represents a data set, with row/column i, j element $[\mathbf{X}]_{ij} = x_{ij}$. We refer to $\mathbf{x}_j$ as a *feature*, *predictor* or *independent variable*. The latter two are more common in statistical literature, the first more common in computer science. Intuitively, a feature refers to a type of

information, of which there are p in this data set. Each element of feature $\mathbf{x}_j$ is a member of an outcome set $x_{ij} \in \mathcal{E}_j$. These p data types may be of any form, quantitative (integer, real or complex), qualitative or categorical (nominal or ordinal) or logical (true or false). However, it is necessary that all outcomes in a feature be of the same type, and additionally of the same unit (inches, degrees celsius, etc) when they are quantitative. The feature space is then the product space $\dot{x}_i \in \mathcal{E}_{\mathbf{x}} = \mathcal{E}_1 \times \cdots \times \mathcal{E}_p$.

12.1 Some Notational Conventions

The 'tilde' notation $\tilde{x}$ or $\tilde{\beta}$ will be used to denote vectors in general, say, $\tilde{x} = (x_1, \ldots, x_n)$ or $\tilde{\alpha} = (\alpha_1, \ldots, \alpha_p)$. The symbols $\dot{x}_i$ and $\mathbf{x}_j$ denote specifically the row and column vectors of a feature matrix $\mathbf{X}$.

In the context of linear algebra, unless otherwise specified an n-dimensional vector is interpreted as an $n \times 1$ column vector. Therefore, given, for example, $\tilde{x} = (x_1, \ldots, x_n)$ and $\tilde{y} = (y_1, \ldots, y_n)$ we may write

$$\tilde{x}^T \tilde{y} = \sum_{i=1}^{n} x_i y_i.$$

As in the previous section, in some cases bold font will denote vectors, for example $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)$.

If $\boldsymbol{A}$ is a matrix, then the i, j element may be written A_{ij} or $[A]_{ij}$, based on whichever seems clearer in the context.

12.2 Structure of Data

The observations are usually sampling units. That is, observation $\dot{x}_i$ is a set of p features outcomes associated with a single sampling unit (person, city, computer image, etc). It is possible that a observation does not contain outcomes for all features. In this case, a special symbol, say $x_{ij} = \texttt{NA}$, is used when the jth feature of sampling unit i is not observed. This becomes a *missing value*, and considerable research has gone into the development of principled methods for dealing with this problem.

12.2.1 Features

As a practical matter, since $\mathbf{X}$ will be subject to algebraic operations, we are forced to regard any element x_{ij} as a number. Usually, this is easily done. A logical feature can translate outcomes *true* or *false* to integers 1 or 0. An ordinal feature $\mathbf{x}_j$ can be converted to a rating scale $1, \ldots, N$, when there are N ordered outcomes in $\mathcal{E}_j$ (eg, 'nonsmoker', 'light smoker', 'heavy smoker'). We may refer to an indicator variable $\mathbf{x}_j$ as one whose outcome set is $\mathcal{E}_j = \{0, 1\}$. This device is often used to indicate the presence or absence of a particular characteristic in the sampling unit (eg 'possesses college degree' = 1). Formally, this type of feature is nominal, but may be used within algebraic operations in a logical manner. While a single indicator variable can represent a nominal feature with only two outcomes, nominal features with $m > 2$ outcomes can be expanded into m (or $m-1$)

indicator variables, by assigning one indicator variable to each outcome:

$$\begin{bmatrix} \text{Red} \\ \text{Red} \\ \text{Blue} \\ \vdots \\ \text{Green} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ \vdots & & \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ \vdots & \\ 0 & 0 \end{bmatrix}. \tag{12.1}$$

Here, a single feature consisting of outcomes $\mathcal{E}_j = \{\text{Red}, \text{Blue}, \text{Green}\}$ shown in (12.1) has been converted to 3, then 2 indicator variables. It is important to note that the 2 column representation is obtained simply by deleting the third column. No information has been lost, as long as we know that the outcome is Green, if it is not Red or Blue, as the definition of the outcome set $\mathcal{E}_j$ tells us. In general, it is preferable from a mathematical point of view to represent an m outcome nominal feature with $m-1$ indicator variables.

12.2.2 Response

Some datasets include a *response variable* $\mathbf{y} = (y_1, \ldots, y_n)$. As for features, the elements are of a common type, with outcomes in outcome set $y_i \in \mathcal{E}_{\mathbf{y}}$. In fact, in all ways $\mathbf{y}$ is a feature, except that is plays a special role in the machine learning application. Alternatively, we may have a data set $\mathbf{X}$ in which one feature $\mathbf{y} = \mathbf{x}_j$ is selected as the response variable, the selection depending on the objective of the application. In this case we take the data set to be $(\mathbf{y}, \mathbf{X})$.

12.3 Feature Distances

It is sometimes important to define a distance d between two vectors observations $\boldsymbol{u} = (u_1, \ldots, u_m)$, $\boldsymbol{v} = (v_1, \ldots, v_m)$. The most natural is Euclidean distance

$$d(\boldsymbol{u}, \boldsymbol{v}) = \sqrt{(u_1 - v_1)^2 + \ldots + (u_m - v_m)^2},$$

but there are often good reasons why alternative distance functions should be considered. A number of mathematical objects are important when considering this problem, particularly the *metric* and the *norm*.

12.3.1 Metrics

The *metric* can be thought of as a generalization of the notion of Euclidean distance, allowing more flexible notions of distance, while retaining the most important properties.

Definition 12.1. Suppose we have real-valued mapping $d : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}$ operating on two observations $\boldsymbol{u}, \boldsymbol{v}$. Then d is a *metric* if

(i) $d(\boldsymbol{u}, \boldsymbol{v}) \geq 0$ (non-negativity);

(ii) $d(\boldsymbol{u}, \boldsymbol{v}) = 0$ if and only if $\boldsymbol{u} = \boldsymbol{v}$ (identifiabilty);

(iii) $d(\boldsymbol{u}, \boldsymbol{v}) = d(\boldsymbol{v}, \boldsymbol{u})$ (symmetry);

(iv) $d(\boldsymbol{u}, \boldsymbol{v}) \leq d(\boldsymbol{u}, \boldsymbol{w}) + d(\boldsymbol{w}, \boldsymbol{v})$ for any $\boldsymbol{w} \in \mathbb{R}^m$ (triangle inequality).

The term *distance function* may be used for mappings satisfying some but not all of these axioms, which otherwise satisfy the intuitive notion of a distance. For example, if (ii) is replaced by (ii)$'$ $d(\boldsymbol{u}, \boldsymbol{v}) = 0$ if $\boldsymbol{u} = \boldsymbol{v}$; then d is a *pseudometric*. If (iii) does not hold then d is a *quasimetric*. A non-symmetric mapping can always be *symmetrized* by taking

$$d^*(\boldsymbol{u}, \boldsymbol{v}) = d(\boldsymbol{u}, \boldsymbol{v}) + d(\boldsymbol{v}, \boldsymbol{u}).$$

Note that a metric multiplied by a positive scalar remains a metric. A real-valued mapping $s : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}$ operating on two observations $\boldsymbol{u}, \boldsymbol{v}$ is a *similarity measure* if it is negatively associated with a distance. □

The similarity measure of Definition 12.1 is not as precisely defined as a metric. Some conventions require that it be non-negative or symmetric. On the other hand, a similarity measure can be simply constructed as $s(\boldsymbol{u}, \boldsymbol{v}) = -d(\boldsymbol{u}, \boldsymbol{v})$, where d is a distance function, and later standardized to be non-negative if needed.

12.3.2 L^p Norms

The magnitude of a vector $\boldsymbol{u} = (u_1, \ldots, u_m)$ in Euclidean space is

$$|\boldsymbol{u}| = \sqrt{u_1^2 + \ldots u_m^2}.$$

Similarly, the Euclidean distance between vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is $|\boldsymbol{u} - \boldsymbol{v}|$. In much the same way that the metric generalizes Euclidean distance, the *norm* generalizes Euclidean magnitude.

Definition 12.2. Suppose we have real-valued mapping $\|\cdot\| : \mathbb{R}^m \mapsto \mathbb{R}$ is a *norm* if for any vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^m$ and scalar $a \in \mathbb{R}$

(i) $\|a\boldsymbol{u}\| = |a|\|\boldsymbol{u}\| \geq 0$ (absolute scalability);

(ii) $\|\boldsymbol{u}\| = 0$ implies $\boldsymbol{u} = \vec{0}$ (identifiabilty);

(iii) $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangle inequality).

Note that $\vec{0} = (0, \ldots, 0)$ is the *zero vector*. It is not necessary to state that $\|\vec{0}\| = 0$ since this is implied by axiom (i). In addition, that $\|\boldsymbol{u}\| \geq 0$ follows from axiom (i) ($\|\boldsymbol{u}\| = \|-\boldsymbol{u}\|$) and axiom (iii) (set $\boldsymbol{v} = -\boldsymbol{u}$).

If axiom (ii) does not hold, then $\|\cdot\|$ is a *semimetric*, which shares all properties of a metric, except that $\|\boldsymbol{u}\| = 0$ does not imply that $\boldsymbol{u} = \vec{0}$. □

The L^p norms are an important class of norms.

Definition 12.3. The L^p norm for $\boldsymbol{u} \in \mathbb{R}^m$ is defined as

$$\|\boldsymbol{u}\|_p = \left[\sum_{i=1}^{m} u_i^p\right]^{1/p},$$

for $p \in (0, \infty)$. In addition, the *supremum norm* (setting $p = \infty$) is defined as

$$\|\boldsymbol{u}\|_\infty = \max_{i=1,\ldots,m} |u_i|.$$

It can be verified that L^p norms are true norms according to Definition 12.2.

Suppose $w_i > 0$, $i = 1, \ldots, m$ are a set of *weights*. The *weighted* L^p norm, denoted L^p_w, is defined as

$$\|\boldsymbol{u}\|_{p,w} = \left[\sum_{i=1}^{m} (w_i u_i)^p\right]^{1/p},$$

for $p \in (0, \infty)$. In addition, the *weighted supremum norm* is defined as

$$\|\boldsymbol{u}\|_{\infty,w} = \max_{i=1,\ldots,m} |w_i u_i|.$$

Weighted L^p_w norms are also true norms. □

12.3.3 Distance Functions

One property of norms proves to be useful in statistical learning. Given any norm $\|\cdot\|$ on $\mathbb{R}^m$ (Definition 12.2), the distance function

$$d(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|$$

is a true metric (Definition 12.1). Many commonly used distance functions are based on norms, including *Euclidean distance*

$$d_{euc}(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_2,$$

Manhattan distance

$$d_{man}(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_1,$$

and *supremum* or *maximum distance*

$$d_{sup}(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_\infty.$$

More variety of distances follow by using weighted L^p_w norms in the same way. An $m \times m$ matrix Σ is *positive definite* if $\boldsymbol{u}^T \Sigma \boldsymbol{u} > 0$ for all nonzero column vectors $\boldsymbol{u}$. In this case Σ is invertible, and Σ^{-1} is also positive definite. Then *Mahalanobis distance* is defined by

$$d_{mah}(\boldsymbol{u}, \boldsymbol{v}) = \left[(\boldsymbol{u} - \boldsymbol{v})^T \Sigma^{-1} (\boldsymbol{u} - \boldsymbol{v})\right]^{1/2},$$

and is a true metric. When this metric is used, Σ is usually a covariance matrix. Note that some conventions refer to Mahalanobis distance as d^2_{mah}. However, d^2_{mah} is not a metric (it does not satisfy the axioms of Definition 12.2). This distinction should be kept in mind.

If $\boldsymbol{u}$ and $\boldsymbol{v}$ are binary vectors, assuming only values in $\{0, 1\}$, then *Hamming distance* is often used, which is equivalent to

$$d_{ham}(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_1,$$

that is, Hamming distance is the L^1 metric applied to binary vectors. It is equivalent to the number of element pairs which differ between $\boldsymbol{u}$ and $\boldsymbol{v}$.

It is also possible to define similarity or distance using a correlation coefficient. A correlation already serves as a similarity measure, and a distance function can be defined as

$$d_{cor}(\boldsymbol{u}, \boldsymbol{v}) = 1 - r(\boldsymbol{u}, \boldsymbol{v})$$

where $r(\boldsymbol{u}, \boldsymbol{v})$ can be any correlation coefficient, including the Pearson and Spearman correlation coefficients, and Kendall's τ.

12.4 Supervised and Unsupervised Learning

Given a set of features $\mathbf{X}$, and possibly a response $\mathbf{y}$, we may define the two main classes of problems in statistical learning. These are distinguished by the presence or absence of a response variable.

In *unsupervised learning*, there is no response variable $\mathbf{y}$. The goal is to uncover relationships or patterns within the observations or features. Perhaps the most common application is *cluster analysis*, in which observations, or possibly features, are divided into *clusters* of similar observations (or features).

In *supervised learning* the objective is to relate the features $\mathbf{X}$ to the response variable $\mathbf{y}$. The formal object is to develop a mapping $\hat{f} : \mathcal{E}_{\mathbf{x}} \mapsto \mathcal{E}_{\mathbf{y}}$, with the property that $\hat{f}(\dot{x}_i) \approx y_i$, in some sense, whether $\mathbf{y}$ is qualitative or quantitative.

The distinction can be seen in Figure 12.1. The MPG and Horsepower ratings of a sample of cars manusfactured in 1973 and 1981 are shown as a scatterplot. The year is indicated by distinct symbols. If we ignore the year, we may note that the points seem to separate into two distinct clusters, and we can imagine a boundary A between them (Figure 12.1). Intuitively, we might conjecture that the two clusters are distinct car styles, perhaps large sedans on one side of the boundary, and smaller economy cars on the other. This is an example of unsupervised learning, since the boundary was constructed without any information beyond the two features MPG and Horsepower.

Next, suppose we wish to develop a rule with which to predict the manufacture year, which then becomes the response variable. We will study methods which, when applied to this data set, might yield a boundary similar to B (Figure 12.1). This yields a mapping $\hat{f}$ which assigns category $\hat{f}(\dot{x}) =$ '1981' if $\dot{x}$ is in the interior of B, and $\hat{f}(\dot{x}) =$ '1973' otherwise. We can assess the accuracy of $\hat{f}$ by systematically comparing $\hat{f}(\dot{x}_i)$ to y_i. If we do, we find that there are two observations with response '1981' outside boundary B and two observations with response '1973' inside, for a total of 4 errors, but all other predictions are correct. Note that to construct this boundary we needed to know the response $\mathbf{y}$. This is a typical example of supervised learning.

12.5 Loss and Risk

The purpose of $\hat{f}$ is to predict a response y given an observation $\dot{x}$, developed from data $(\mathbf{y}, \mathbf{X})$ using the types of methodologies we will discuss below.

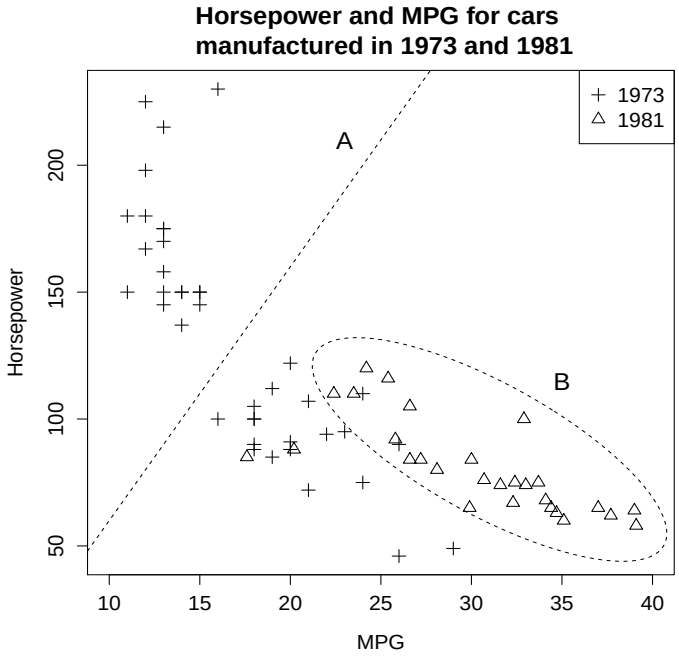

Figure 12.1: Sample of cars manufactured in 1973 and 1981 (Section 12.4).

Note that $\mathbf{y}$ may be qualitative or quantitative. Up to a point, the principles of supervised learning do not depend on this distinction, but eventually, the difference becomes unavoidable. In either, case, it is assumed that a response y obeys a distribution $f(y \mid \dot{x})$ conditional on its associated feature. When a new feature $\dot{x}$ is presented, we form a prediction $\hat{y} = \hat{f}(\dot{x})$ for y. Recognizing that a prediction $\hat{f}(\dot{x})$ is subject to some error, we define a *loss function* $L(y, \hat{y})$, which is our assessment of the cost of prediction error. This can vary with the goal of the predictor, and a variety of loss functions might be considered. This is because we might view one type of error as more consequential than another (a false negative would be of greater consequence for a fire alarm than a false positive). Given a probability model, we define *risk*:

$$R(\dot{x}) = E[L(y, \hat{f}(\dot{x}))],$$

which is the expected loss for feature $\dot{x}$. The goal of a predictor is to minimize risk, the predictor which achieves this will depend on the loss function.

We first assume that $\mathbf{y}$ is quantitative. In this case we define the model

$$y = f(\dot{x}) + \epsilon. \tag{12.2}$$

Here, ϵ is a random error, assumed to have a mean $E[\epsilon] = 0$. Furthermore, ϵ is often assumed to be normally distributed, that is, $\epsilon \sim N(0, \sigma^2)$. The conditional density of response y is then $y \sim N(f(\dot{x}), \sigma^2)$. There are good reasons for this assumption, but the basic ideas do not depend on this. Of course, it is possible that σ^2 depends on $\dot{x}$. The usual practice is to first develop a theory of statistical modeling based on the assumption of constant σ^2, then to modify these models for more general cases. An example of a model with varying σ^2 (logistic regression) will be discussed in Chapter 13.9.

The mapping f is not known, so must be estimated by $\hat{f}$. We now formally state the problem. We are given data $(\mathbf{y}, \mathbf{X})$. Depending on the methodology, $\hat{f}$ is to be chosen from some class of functions $\hat{f} \in \mathcal{F}$. Clearly, $\hat{f}$ should be close to the f given in (12.2). We can achieve this by systematically testing candidate functions $\hat{f} \in \mathcal{F}$ using the data. Assuming ϵ is not too large, if $\hat{f} \approx f$, then we would expect

$$y_i \approx \hat{f}(\dot{x}_i), \;\; 1, \ldots, n.$$

All elements of this approximation are observable, so we develop an aggregate *goodness of fit* measure. The most commonly used is the *error sum of squares*:

$$SSE = \sum_{i=1}^{n} (y_i - \hat{f}(\dot{x}_i))^2 = \sum_{i=1}^{n} e_i^2,$$

where $e_i = y_i - \hat{f}(\dot{x}_i)$ are the *residuals*. Note that SSE is also known as the *residual sum of squares* RSS. This corresponds to a loss function $L(x, y) = (x - y)^2$.

The *least squares* fit $\hat{f}$ is then

$$\hat{f} = \text{argmin}_{f^* \in \mathcal{F}} SSE[f^*],$$

where we write $SSE[f^*]$ to emphasis the dependence on the sum of squares on f^*. By convention, when we write SSE alone, this refers to the minimum possible value over $\mathcal{F}$.

We now ask a crucial question. What will SSE be if we are correct, that is, if $\hat{f} = f$? In this case, $e_i = y - f(\dot{x}) = \epsilon_i$, where ϵ_i is the true error term given in (12.2). If we assume that $var\,[\epsilon_i] = \sigma^2$, whether or not ϵ_i is normally distributed, then

$$SSE = \sum_{i=1}^{n} \epsilon_i^2,$$

and the *mean squared error* MSE will be

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \epsilon_i^2 \approx \sigma^2. \tag{12.3}$$

This means the object is not to make the MSE as close to zero as possible, rather, it is to make it as close to σ^2 as possible.

To see the problem suggested here, suppose we attempt a simple regression model with only two observations. It will then be possible to find a line which passes exactly through these points, and we will have $MSE = 0$. This, of course, does not mean that the true model is error free, rather, it means that our modeling method is not well conceived. In particular, our model space $\mathcal{F}$ is too large. In fact, there is a special term for this. We say that a model which is sufficiently flexible to force $\hat{f}(\dot{x}_i) = y_i$ for all i is a *saturated model* (this idea is discussed further in Chapters 7 and 13.9). To achieve this, we only need let $\mathcal{F}$ be 'the set of all functions conceivable', and we will achieve $MSE = 0$. More realistically, it is always possible that the model space $\mathcal{F}$ is large and rich enough that the process of minimizing MSE leads to an underestimate of σ^2, a problem usually refered to as *overfitting*.

How do we overcome this problem, especially if we don't know the value of σ^2, which is usually the case? We first need to recognize that we really have two estimation problems. We need to

estimate f using $\hat{f} \in \mathcal{F}$, but we also need to estimate the appropriate level of complexity for $\mathcal{F}$. That is, we are estimating both $\mathcal{F}$ and $\hat{f}$. This process is referred to as *model selection*.

Of course, this problem is related to the estimation of σ^2, and a general solution is expressible in those terms. Suppose we set our goal not as selecting $\hat{f}$ which minimizes MSE, but as the function which minimizes

$$MSE_{test} = E\left[(y' - \hat{f}(\dot{x}'))^2\right]$$

where $(y', \dot{x}')$ is an response/observation pair not previously sampled (but sampled under identical conditions). Then, omitting some details, we have, given model (12.2),

$$\begin{aligned} MSE_{test} &= E[(y' - f(\dot{x}') + f(\dot{x}') - \hat{f}(\dot{x}'))^2] \\ &= E[\epsilon^2 + 2\epsilon(f(\dot{x}') - \hat{f}(\dot{x}')) + (f(\dot{x}') - \hat{f}(\dot{x}'))^2] \\ &= E[\epsilon^2] + E[(f(\dot{x}') - \hat{f}(\dot{x}'))^2] \\ &= \sigma^2 + E[(f(\dot{x}') - \hat{f}(\dot{x}'))^2]. \end{aligned} \tag{12.4}$$

The second term of (12.4) is positive, and also approaches 0 as $\hat{f}$ becomes more accurate, in which case MSE_{test} yields an estimate of σ^2. Therefore, finding $\hat{f}$ which minimizes MSE_{test} is a better strategy than minimizing the MSE defined in (12.3). In fact, this is the approach which minimizes risk based on squared error loss.

12.6 Cross-Validation

The problem now is the estimation of MSE_{test}. We can think of this as a two stage process. First, we build a predictor $\hat{f}$ using *training data* $(\mathbf{y}, \mathbf{X})$. Then we estimate MSE_{test} distinct *test data* $(\mathbf{y}', \mathbf{X}')$. If $\hat{f}$ is the predictor fit using the training data, then

$$MSE_{test} \approx \frac{1}{n}\sum_{i=1}^{n'}(y_i' - \hat{f}(\dot{x}_i'))^2,$$

where $(y_i', \dot{x}_i')$, $i = 1, \ldots, n'$, are the response/observation pairs from the test data set. It is important to note that the data used to build the predictor $\hat{f}$ is independent of the data used to test the predictor's accuracy. The degree to which a predictors' accuracy is overestimated by failing to do this can be surprisingly large, particularly with smaller data sets.

So, where does the test data come from? In medical studies, collecting new data to test a previously developed model is inevitable, for any model which shows promise. Absent this, we may take the point of view that we already have test data. All we need do is divide our current data into training and test data sets, preferably at random. At this point, we may then refer to the MSE calculated from training data as MSE_{train}.

We may then use the following approach.

Algorithm 12.1. For a given data set $(\mathbf{y}, \mathbf{X})$ take the following steps:

1. Define a sequence of model spaces $\mathcal{F}_1, \ldots, \mathcal{F}_K$.
2. Calculate $\hat{f}_i$ minimizing MSE_{train} on model space $\mathcal{F}_i$, $i = 1, \ldots, K$.
3. Estimate MSE_{test} for each predictor $\hat{f}_i$, $i = 1, \ldots, K$.

4. Select the predictor with the smallest MSE_{test}.

This simple partition method is sound, but may be subject to considerable variability, and the predictor may depend considerably on the particular training/test partition. One alternative is *cross-validation* (CV). Suppose we delete the first response/observation pair from the data set, then fit a predictor $\hat{f}^{(-1)}$ using the remaining data. Then we may expect

$$E[(y_1 - \hat{f}^{(-1)}(\dot{x}_1))^2] \approx MSE_{test}.$$

Doing this for each observation yields the CV estimate

$$MSE_{CV} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}^{(-i)}(\dot{x}_i))^2, \tag{12.5}$$

and we expect $MSE_{CV} \approx MSE_{test}$.

More generally, we have k-fold cross validation. The sample is divided into k groups, or *folds*. The first group is used as test data for a model fit with the remaining (training) data, yielding $MSE_{test}(1)$. This is repeated for each group, and the cross-validated MSE is taken to be the average

$$MSE_{CV}[k] = \frac{1}{k}\sum_{i=1}^{k} MSE_{test}(i).$$

When $k = n$, we have (12.5), commonly referred to as *leave-one-out* CV (LOOCV).

In this way Algorithm 12.1 is altered accordingly:

Algorithm 12.2. For a given data set $(\mathbf{y}, \mathbf{X})$ and some fixed k take the following steps:

1. Define a sequence of model spaces $\mathcal{F}_1, \ldots, \mathcal{F}_K$.
2. Calculate $MSE_{CV}[k]$ for each model space $i = 1, \ldots, K$.
3. Select the model space yielding the smallest $MSE_{CV}[k]$. Use this to construct the predictor.

12.7 Bias and Variance

It is worth decomposing equation (12.4) a little further. First note that an estimator $\hat{\theta}$ of any parameter θ is *unbiased* if $E[\hat{\theta}] = \theta$ (otherwise they are *biased*). Despite what we might expect, not all estimators are unbiased, including many widely used ones. The *bias* of an estimate is defined as

$$Bias[\hat{\theta}] = E[\hat{\theta}] - \theta.$$

Then for squared error loss function $L(\theta, \hat{\theta})$, the risk is, after some algebra,

$$\begin{aligned} R(\theta) &= E\left[(\hat{\theta} - \theta)^2\right] \\ &= E\left[\left(\hat{\theta} - E[\hat{\theta}] + E[\hat{\theta}] - \theta\right)^2\right] \\ &= var[\hat{\theta}] + Bias[\hat{\theta}]^2. \end{aligned} \tag{12.6}$$

It might seem a simple matter to subtract the bias from $\hat{\theta}$ to yield an unbiased estimated, thereby reducing risk. Of course, we can't do this if the bias depends on θ, which is unknown, and this is often the case.

In statistical inference we are sometimes confronted with the bias/variance tradeoff suggested by (12.6). Suppose we implement Algorithm 12.1 or 12.2. We first note that MSE_{test} is an estimate of risk. Then, we will often find that the sequence of model spaces $\mathcal{F}_1, \ldots, \mathcal{F}_K$ represents a gradual decrease in complexity. More complex models tend to have less bias, due to their greater flexibility, but also more variance, again due to their greater flexibility. As complexity decreases, the variance term in (12.6) decreases, while the bias term increases. Hopefully, at some point within the sequence we will find a 'sweet spot', where the optimal tradeoff between variance and bias yields the minimum MSE_{test} or, approximately, risk.

12.8 Model Selection for Classifiers

The approach of Section 12.6 can be used for classifiers. We are using a different loss function, usually $L(y, \hat{y}) = I\{y \neq \hat{y}\}$, so that risk is now

$$R(\hat{f}(\dot{x})) = P(y \neq \hat{f}(\dot{x}))$$

for a new response/observation pair $(y, \dot{x})$. Given training data, the risk can be estimated as the observed error rate

$$CE_{train} = \frac{1}{n} \sum_{i=1}^{n} \{y_i \neq \hat{f}(\dot{x}_i)\}.$$

Of course, this measure suffers from the same defects as MSE_{train}, that is, it is subject to overfitting, and may overestimate the predictor's accuracy. We may use CV and Algorithms 12.1-12.2 in much the same way as described in Section 12.6, except that observed classification error is used in place of mean squared error. Then the quantities CE_{test}, CE_{CV} and $CE_{CV}[k]$ follow in much the same way.

Chapter 13

Bayes Theorem and Classification

If we are given a conditional probability $P(E \mid A)$ we often would like to "reverse the order" of the events to obtain $P(A \mid E)$. To do this we use *Baye's theorem*

Theorem 1. For two events A and E, with $P(E) > 0$, we have

$$P(A \mid E) = P(E \mid A)\frac{P(A)}{P(E)}. \tag{13.1}$$

□

Proof. The Equation (13.1) is proven with the following argument:

$$\begin{aligned} P(A \mid E) &= \frac{P(AE)}{P(E)} \\ &= \frac{P(E \mid A)P(A)}{P(E)}. \end{aligned}$$

□

The following definition, though straightforward, is quite important to understanding the current chapter.

Definition 13.1. In the context of Baye's theorem $P(A)$ is the *prior probability* of A, and $P(A \mid E)$ is the *posterior probability* of A given information E. □

The following is a useful variation of Baye's theorem.

Theorem 2. Suppose events $A_1, \ldots, A_n$ is a partition of sample space S, that is, the events are mutually exclusive with

$$S = \cup_{i=1}^{n} A_i.$$

For any $1 \leq i \leq n$

$$\begin{aligned} P(A_i \mid E) &= \frac{P(E \mid A_i)P(A_i)}{P(E)} \\ &= \frac{P(E \mid A_i)P(A_i)}{P(E \mid A_1)P(A_1) + \ldots + P(E \mid A_n)P(A_n)}. \end{aligned} \tag{13.2}$$

□

Proof. Equation 13.2 follows from the law of total probability. □

Example 13.1. Suppose a test for a certain infection is evaluated by administering the test to 50 patients with the infection, and 100 patients known to be without the infection (control patients). The test was positive for 49 of the 50 infected patients and positive for 4 of the 100 control patients. Let

$$\begin{aligned} T &= \{ \text{ Patient tests positive } \} \\ D &= \{ \text{ Patient has infection } \} \end{aligned}$$

From the above data we can estimate directly

$$\begin{aligned} P(T \mid D) &= 49/50 \\ P(T^c \mid D) &= 1/50 \\ P(T \mid D^c) &= 4/100 \\ P(T^c \mid D^c) &= 96/100. \end{aligned}$$

Thus, from the data we get directly

$$P(T \mid D) = \text{Probability of testing positive when infected}$$

and

$$P(T \mid D^c) = \text{Probability of testing positive when not infected}$$

but what is of ultimate interest are the probabilities

$$P(D \mid T) = \text{Probability of being infected when testing positive}$$

and

$$P(D \mid T^c) = \text{Probability of being infected when not testing positive}$$

since this is the quantity which is clinically relevant. We use Baye's Theorem to calculate these probabilities, setting

$$\begin{aligned} A_1 &= D \\ A_2 &= D^c. \end{aligned}$$

Then

$$\begin{aligned} P(D \mid T) &= \frac{P(T \mid D)P(D)}{P(T \mid D)P(D) + P(T \mid D^c)P(D^c)} \\ &= \frac{(49/50)P(D)}{(49/50)P(D) + (4/100)P(D^c)} \end{aligned}$$

and

$$\begin{aligned} P(D \mid T^c) &= \frac{P(T^c \mid D)P(D)}{P(T^c \mid D)P(D) + P(T^c \mid D^c)P(D^c)} \\ &= \frac{(1/50)P(D)}{(1/50)P(D) + (96/100)P(D^c)}. \end{aligned}$$

Note that in order to evaluate these probabilities we need to know $P(D)$ which was not obtained from the original experiment. This should be the case, since if $P(D) = 0$ (*i.e.*, the infection is nonexistent) we would also expect both $P(D \mid T) = P(D \mid T^c) = 0$. □

13.1 Odds

The term *odds* is synonymous with probability, and is formally defined as follows:

Definition 13.2. For a given event A we define the *odds* to be

$$Odds(A) = \frac{P(A)}{P(A^c)} = \frac{P(A)}{1 - P(A)}.$$

□

If I roll a die, the probability of getting a six is 1/6, but the odds are 1/5. Mathematically, the odds and the probability of an event A are equivalent, since we can calculate the odds from the probability, as well as the probability from the odds:

$$P(A) = \frac{Odds(A)}{1 + Odds(A)}.$$

In particular, if A is certain to occur then

$$\begin{aligned} P(A) &= 1 \\ Odds(A) &= \infty \end{aligned}$$

and if A is certain to not occur then

$$\begin{aligned} P(A) &= 0 \\ Odds(A) &= 0. \end{aligned}$$

We can also define the *conditional odds* of A given E.

Definition 13.3. The *conditional odds* of A given E is defined as

$$Odds(A \mid E) = \frac{P(A \mid E)}{P(A^c \mid E)} = \frac{P(A \mid E)}{1 - P(A \mid E)}.$$

□

The conditional odds leads to a particularly intuitive form of Baye's theorem.

Theorem 3. The conditional odds of A given E may be expressed

$$Odds(A \mid E) = \frac{P(E \mid A)}{P(E \mid A^c)} \times Odds(A). \tag{13.3}$$

□

Proof. Equation (13.3) is proven with the following argument:

$$\begin{aligned} Odds(A \mid E) &= \frac{P(A \mid E)}{P(A^c \mid E)} \\ &= \frac{P(E \mid A)P(A)}{P(E)} \times \frac{P(E)}{P(E \mid A^c)P(A^c)} \\ &= \frac{P(E \mid A)}{P(E \mid A^c)} \times \frac{P(A)}{P(A^c)} \\ &= \frac{P(E \mid A)}{P(E \mid A^c)} \times Odds(A). \end{aligned}$$

□

13.2 The Bayesian Model

Under the **Bayesian model** we are interested in the probability of a hypothesis A, or more specifically, the effect on this probability of the introduction of information or evidence E. There may be a well known prevalence of a certain condition (hypothesis A) among a population. For any given patient entering a clinic, this prevalance may be $P(A)$. A diagnostic test is then done. Let E be the event that this test is positive. We are now no longer interested in $P(A)$, but in $P(A \mid E)$ or $P(A \mid E^c)$, depending on the outcome of the test.

Based on an evaluation of the accuracy of the test, we may know $P(E \mid A)$ and $P(E \mid A^c)$. Examining Equation (13.3), we define the *likelihood ratio* as follows:

Definition 13.4. When considering the odds of an event A given evidence E, the *likelihood ratio* is given by

$$LR = \frac{P(E \mid A)}{P(E \mid A^c)},$$

from which we get, as a reexpression of Theorem 3,

$$Odds(A \mid E) = LR \times Odds(A). \tag{13.4}$$

We refer to $Odds(A)$ as the *prior odds* and to $Odds(A \mid E)$ as the *posterior odds*. □

The relationship between the prior and posterior odds is the same as that between the prior and posterior probability. However, Equation (13.4) very neatly captures the ability of the evidence to alter our assessment of the probability of a hypothesis in a way which does not depend on the prior probability.

Example 13.2. We will express the previous Example 13.1 in terms of odds of having the infection. If a patient tests positive, the odds are adjusted by the formula

$$\begin{aligned} Odds(D \mid T) &= \frac{P(T \mid D)}{P(T \mid D^c)} \times Odds(D) \\ &= \frac{49/50}{4/100} \times Odds(D) \\ &= 24.5 \times Odds(D) \end{aligned}$$

so that testing positive *increases* the odds of having the infection by a factor of 24.5.

If the patient tests negative, the odds are adjusted by the formula

$$\begin{aligned} Odds(D \mid T^c) &= \frac{P(T^c \mid D)}{P(T^c \mid D^c)} \times Odds(D) \\ &= \frac{1/50}{96/100} \times Odds(D) \\ &= \frac{1}{48} \times Odds(D) \end{aligned}$$

so that testing negative *decreases* the odds of having the infection by a factor of 48.

We are therefore in a better position to evaluate the accuracy of the test when the problem is expressed in terms of odds. □

Example 13.3. Suppose blood collected at a crime scene is typed for DNA. A genotype if found which is estimated to occur in the population with a frequency of p. A suspect is similarly typed and found to have the same genotype. Suppose

$$\begin{aligned} A &= \{ \text{ Suspects blood is that found at the crime scene } \} \\ E &= \{ \text{ Suspect has the same genotype as blood found at crime scene } \} \end{aligned}$$

Then the likelihood ratio is constructed by noting that

$$\begin{aligned} P(E \mid A) &= 1 \\ P(E \mid A^c) &= p \end{aligned}$$

giving

$$\begin{aligned} LR &= \frac{P(E \mid A)}{P(E \mid A^c)} \\ &= \frac{1}{p} \end{aligned}$$

so that the odds that the blood is the same is adjusted by

$$\begin{aligned} Odds(A \mid E) &= LR \times Odds(A) \\ &= \frac{1}{p} \times Odds(A) \end{aligned}$$

We usually have no way of directly evaluating $Odds(A)$. We can only describe how the evidence changes the odds. If it were established without doubt that the suspect was not at the crime scene by other evidence then we would have

$$Odds(A) = 0$$

and

$$Odds(A \mid E) = 0$$

for any value of LR. If guilt were established with absolute certainty then

$$Odds(A) = \infty$$

and

$$Odds(A \mid E) = \infty.$$

for any value of LR.

Now, suppose the genotype does not match. (That is, E^c occurs). The likelihood ratio is now calulated from

$$\begin{aligned} P(E^c \mid A) &= 0 \\ P(E^c \mid A^c) &= 1 - p \end{aligned}$$

giving $LR = 0$ so that

$$\begin{aligned} Odds(A \mid E) &= LR \times Odds(A) \\ &= 0 \end{aligned}$$

for any $Odds(A)$. □

13.3 The Fallacy of the Transposed Conditional

In the previous example suppose we set $p = 1/100$. We could then say

$$P(E \mid A^c) = 1/100$$

which is the probability of a genotype match if the suspect is not guilty. A common error is to *transpose the conditional* which yields (incorrectly)

$$P(A^c \mid E) = 1/100.$$

This statement says that the probability that the suspect is not guilty is 1/100 if a match occurs. After some algebra we then get

$$\begin{aligned} P(A \mid E) &= 1 - P(A^c \mid E) \\ &= 99/100 \end{aligned}$$

which says that, given a match, the probability that the suspect is guilty is 99/100, which cannot be concluded from the evidence. The odds of guilt given the evidence depends on the prior odds. This is often referred to as the *prosecutor's fallacy*.

13.4 Diagnostic Testing - Basic Definitions

A common problem in medical research is the evaluation of the accuracy of diagnostic tests. This can be framed in the context of probability theory. In the simplest case, a diagnostic test is either positive (in which case the patient is predicted to have the condition being tested) or negative (in which case the patient is predicted to not have the condition being tested). There are 4 events of interest:

$$\begin{aligned} O_- &= \{ \text{ the patient does not have the condition } \} \\ O_+ &= \{ \text{ the patient has the condition } \} \\ T_- &= \{ \text{ the patient tests negative } \} \\ T_+ &= \{ \text{ the patient tests positive } \} \end{aligned}$$

Clearly, $O_-^c = O_+$ and $T_-^c = T_+$, so that $P(O_-) + P(O_+) = 1$ and $P(T_-) + P(T_+) = 1$.

Conditional probabilities and Bayes theorem can be very useful in developing a probabilistic model for these outcomes, and a widely used terminology has been developed:

$$\begin{aligned} \text{sensitivity } (\textit{sens}) &= P(T_+ \mid O_+) \\ \text{specificity } (\textit{spec}) &= P(T_- \mid O_-) \\ \text{positive predictive value (PPV)} &= P(O_+ \mid T_+) \\ \text{negative predictive value (NPV)} &= P(O_- \mid T_-) \\ \text{prevalance } (\textit{prev}) &= P(O_+). \end{aligned} \tag{13.5}$$

Two more related definitions are sometimes used:

$$\begin{aligned} \text{true discovery rate (TDR)} &= \textit{sens} \\ \text{false discovery rate (FDR)} &= P(T_+ \mid O_-) = 1 - \textit{spec}. \end{aligned}$$

In an evaluation study, a diagnostic test will typically be administered to subjects with known outcomes, which will allow sensitivity and specificity to be estimated. However, when used in a clinical setting, the outcomes will not be known. These are to be predicted based on the test result, so it will be PPV and NPV which are more relevant. These quantities can be related to sensitivity, specificity and prevalence using Baye's Theorem:

$$\begin{aligned} PPV &= P(O_+ \mid T_+) \\ &= \frac{P(T_+ \mid O_+)P(O_+)}{P(T_+ \mid O_+)P(O_+) + P(T_+ \mid O_-)P(O_-)} \\ &= \frac{sens \times prev}{sens \times prev + (1 - spec) \times (1 - prev)} \end{aligned} \tag{13.6}$$

and

$$\begin{aligned} NPV &= P(O_- \mid T_-) \\ &= \frac{P(T_- \mid O_-)P(O_-)}{P(T_- \mid O_-)P(O_-) + P(T_- \mid O_+)P(O_+)} \\ &= \frac{spec \times (1 - prev)}{spec \times (1 - prev) + (1 - sens) \times prev}. \end{aligned} \tag{13.7}$$

It is important to understand the degree to which PPV and NPV depend on prevalence. We have already seen in Example 13.1 that if, for example, $prev = 0$ we would necessarily have $PPV = 0$, no matter what sensitivity and specificity are. On the other hand, sensitivity and specificity do not depend on prevalence, and this distinction is an important one.

13.4.1 Diagnostic Tests and Contingency Tables

The outcomes of a study used to evaluate a diagnostic test can be summarized in Table 1 below,

Table 1: Outcomes of diagnostic testing

		Condition	
		Positive	Negative
Test	Positive	TP	FP
	Negative	FN	TN

where

$$\begin{aligned} \text{TP} &= T_+ \cap O_+ = \text{True Positive} \\ \text{FP} &= T_+ \cap O_- = \text{False Positive} \\ \text{TN} &= T_- \cap O_- = \text{True Negative} \\ \text{FN} &= T_- \cap O_+ = \text{False Negative}. \end{aligned} \tag{13.8}$$

Table 1 can be interpreted as a contingency table, with numerical entries TP, FP, TN, FN giving the counts of subjects in each category. These can be used to estimate all important quantities. If

we let $N = TP + FP + TN + FN$ (the total number of entries in Table 1) we can calculate the *marginal probabilities*:

$$\begin{aligned} P(O_-) &= \frac{FP + TN}{N} \\ P(O_+) &= \frac{TP + FN}{N} \\ P(T_-) &= \frac{FN + TN}{N} \\ P(T_+) &= \frac{TP + FP}{N}, \end{aligned} \tag{13.9}$$

and the important diagnostic quantities

$$\begin{aligned} prev &= \frac{TP + FN}{N} = P(O_+) \\ sens &= \frac{TP}{TP + FN} = P(T_+ \mid O_+) \\ spec &= \frac{TN}{TN + FP} = P(T_- \mid O_-) \\ PPV &= \frac{TP}{TP + FP} = P(O_+ \mid T_+) \\ NPV &= \frac{TN}{TN + FN} = P(O_- \mid T_-). \end{aligned} \tag{13.10}$$

However, the prevalence must be very carefully interpreted. If we calculate *prev* directly from Table 1, we obtain the prevalence of an outcome within the study population, which may have no relationship to the prevalence in any given clinical population. This would be especially true if the study was designed to ensure a large enough sample of disease positive subjects to accurately evaluate the test. In such cases, we would expect *prev* to be much higher than it would be in a clinical population.

Therefore, it is important to understand that it is always possible, and usually preferable, to calculate prevalence independently of sensitivity and specificity. In particular, if we are using a study such as that represented by Table 1 we would use equations (13.10) to estimate *sens* and *spec* but not *prev*, PPV or NPV. Instead, we would use an independent estimate of *prev* which more accurately estimates the prevalance within the clinical population of interest, and then use (13.6)-(13.7) with that value of *prev*.

In summary, the important question is whether or not the subjects used in Table 1 are representative of the population in which the test is to be applied, in terms of the relative frequencies of outcomes O_+ and O_-. The values of *prev*, PPV and NPV calculated by equations (13.10) would be interpretable only if this is the case.

13.4.2 The Use of Odds in the Evaluation of Diagnostic Tests

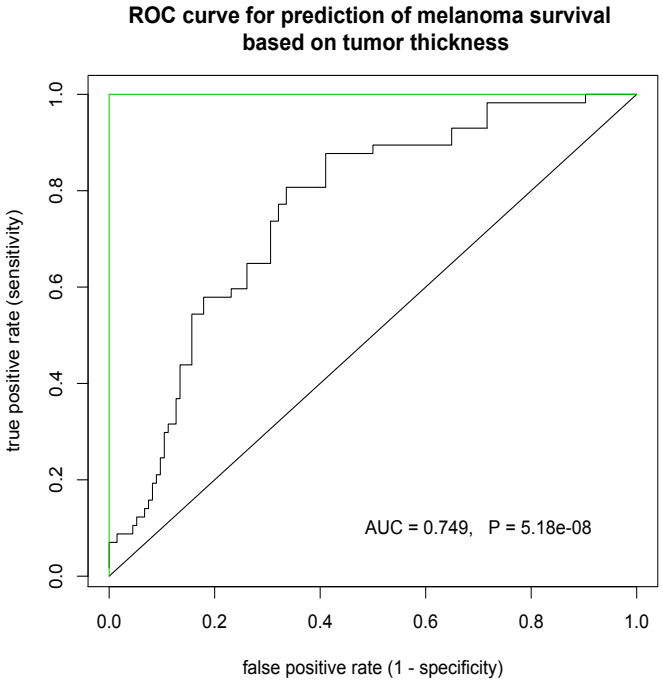

Figure 13.1: ROC curve for prediction of melanoma cancer survival based on melanoma *thickness* variable (tumor thickness in mm). The AUC is given, as well as the p-value for a WIlcoxon rank sum test for group homogeneity of risk score distributions. The green lines represent a "perfect" classifier, with sensitivity and specificity both equal to one. The diagonal identity line represents a noninformative risk score of $AUC = 0.5$.

In the absence of a reliable estimate of prevalence, the accuracy of a diagnostic test can be expressed using odds, as shown above. Using the previous terminology we have

$$LR = \frac{P(T_+ \mid O_+)}{P(T_+ \mid O_-)} = \frac{\text{sensitivity}}{\text{1 - specificity}}$$

so that

$$Odds(O_+ \mid T_+) = LR \times Odds(O_+).$$

Then $Odds(O_+)$ is the prevalence expressed as odds, and the predicitive ability of the test can be expressed using only the sensitivity and specificity.

Note that we can also assess the accuracy of a negative test outcome. In this case we can distinguish between the LR for a positive test outcome LR_+ and the LR for a negative test outcome LR_-:

$$LR_+ = LR \text{ as defined above } LR_- = \frac{P(T_- \mid O_+)}{P(T_- \mid O_-)} = \frac{\text{1 - sensitivity}}{\text{specificity}}$$

so that

$$\begin{aligned} Odds(O_+ \mid T_+) &= LR_+ \times Odds(O_+) \\ Odds(O_+ \mid T_-) &= LR_- \times Odds(O_+). \end{aligned}$$

Example 13.4. Studies into the accuracy of a diagnostic test often proceed by pairing the test with a *gold standard* in a study group of size N, the latter assumed to be perfectly accurate. In this case, we can estimate sensitivity and specificity. After the study we would construct a contingency table like the following ($N = 1000$):

Table 2: Outcomes of diagnostic testing for Example 13.4

		Condition	
		Positive	Negative
Test	Positive	30	110
	Negative	10	850

We have, using equations (13.10), ($TP = 30$, for example):

$$\begin{aligned} sens &= 30/40 = 0.75 \\ spec &= 850/960 \approx 0.885 \\ LR_+ &= (30/40)/(1 - 850/960) \approx 6.545 \\ LR_- &= (1 - 30/40)/(850/960) \approx 0.282 \end{aligned}$$

and the Bayes model gives

$$\begin{aligned} Odds(O_+ \mid T_+) &\approx 6.545 \times Odds(O_+) \\ Odds(O_+ \mid T_-) &\approx 0.282 \times Odds(O_+). \end{aligned}$$

A positive test result increases the odds of a positive outcome by a factor of 6.545, while a negative test result decreases the odds of a positive outcome by a factor of 0.282.

Next, if we calculate $prev$, PPV and NPV dircetly from the contingency table, using equations (13.10). we would have

$$\begin{aligned} prev &= (10 + 30)/1000 = 0.04 \\ PPV &= 30/140 \approx 0.214 \\ NPV &= 850/860 \approx 0.988. \end{aligned}$$

The values of PPV and NPV assume a prevalance of 4%, estimated directly from tha data. Suppose the true prevalence was 2%. We would then use (13.6)-(13.7) with $prev = 0.02$ and the estimates of $sens$ and $spec$ obtained from the data (remember that these quantities do not depend

on the prevalence). This gives

$$\begin{aligned} PPV &= \frac{sens \times prev}{sens \times prev + (1 - spec) \times (1 - prev)} \\ &= \frac{0.75 \times 0.02}{0.75 \times 0.02 + (1 - 0.885) \times (1 - 0.02)} \\ &\approx 0.117 \end{aligned}$$

and

$$\begin{aligned} NPV &= \frac{spec \times (1 - prev)}{spec \times (1 - prev) + (1 - sens) \times prev} \\ &= \frac{0.885 \times (1 - 0.02)}{0.885 \times (1 - 0.02) + (1 - 0.75) \times 0.02} \\ &\approx 0.994. \end{aligned}$$

Reducing the prevalence by 1/2 results in a reduction in PPV of almost the same magnitude (verify that if we use $prev = 0.04$ in equations (13.6)-(13.7) we reproduce the values of PPV and NPV obtained using equations (13.10)).

Using either method, that PPV is much smaller than sensitivity is typical, and is due to the fact that PPV depends on the prevalence. Expecting the two to be equal is an example of the 'prosecutor's fallacy', since one is obtained from the other by transposing the conditional.

Note also that NPV is quite large. This is a function both of the ability of the test to rule out a positive outcome (measured by specificity) and of the relatively small prevalence. This means NPV would be smaller when the test is confined to a higher risk population.

13.5 The Odds Ratio

Consider the following events

$$\begin{aligned} O_- &= \{ \text{ the patient does not have the condition } \} \\ O_+ &= \{ \text{ the patient has the condition } \} \\ G_1 &= \{ \text{ the patient is in Group 1 } \} \\ G_2 &= \{ \text{ the patient is in Group 2 } \}. \end{aligned}$$

Typically, we are interested in comparing

$$P(O_+ \mid G_1) \text{ and } P(O_+ \mid G_2).$$

Perhaps the obvious comparison method is to examine the difference:

$$\Delta = P(O_+ \mid G_1) - P(O_+ \mid G_2).$$

This will be, sometimes, a reasonable approach, but will not work well when the probabilities are small. Alternatively, we have the *relative risk*

$$RR = \frac{P(O_+ \mid G_1)}{P(O_+ \mid G_2)}$$

and the *odds ratio* (OR)

$$OR = \frac{Odds(O_+ \mid G_1)}{Odds(O_+ \mid G_2)} = \frac{P(O_+ \mid G_1)/(1 - P(O_+ \mid G_1))}{P(O_+ \mid G_2)/(1 - P(O_+ \mid G_2))}.$$

The OR has an interesting property, in that events defining it may be transposed, that is

$$OR = \frac{Odds(G_1 \mid O_+)}{Odds(G_1 \mid O_-)},$$

so that the OR does not depend on the marginal probabilities (that is, the prevalences). For some applications, this is a considerable advantage, for the reasons discussed earlier in this chapter.

13.6 Bayes Classifiers

We now consider the problem of classification. We have, as before, responses and features $\mathbf{y}$ and $\mathbf{X}$. But now, $\mathbf{Y}$ is qualitative, and the predictor function $\hat{f}(\dot{x})$ now assigns a predicted class from $\mathcal{E}_\mathbf{y}$ to each feature $\dot{x}$. The additive error model (12.2) is no longer applicable. Instead, either a predicted class is correct, or an incorrect class is predicted. Suppose there are m classes in $\mathcal{E}_\mathbf{y}$, which we can always label $1, \ldots, m$. If there is any information in the features $\dot{x}$ which is able to distinguish between classes, then there must be conditional densities $f(\dot{x} \mid y = j)$, $j \in 1, \ldots, m$ which differ noticeably from each other. We can then use Bayes theorem to get conditional distribution:

$$P(y = j \mid \dot{x}) = \frac{f(\dot{x} \mid y = j)\pi_j}{f(\dot{x})} \tag{13.11}$$

where π_j is the prior probability of class j (the prevalance of class j in the population of interest), and

$$f(\dot{x}) = \sum_{j=1}^{m} f(\dot{x} \mid y = j)\pi_j$$

is the total probability distribution of $\dot{x}$. Suppose we use loss function:

$$L(y, \hat{f}(\dot{x})) = I\{y \neq \hat{f}(\dot{x})\},$$

that is, the loss is one if and only if the classification is incorrect. Then given feature $\dot{x}$, the minimum risk classifier, refered to as the *Bayes classifier* can be shown to be

$$\hat{f}(\dot{x}) = \text{argmax}_{j \in \mathcal{E}_\mathbf{y}} P(y = j \mid \dot{x}) = \text{argmax}_{j \in \mathcal{E}_\mathbf{y}} f(\dot{x} \mid y = j)\pi_j, \tag{13.12}$$

noting that the denominator in (13.11) does not depend on class type j. Of course, we may choose to define a more complex loss function. For example, if the true class is 1, then is may be a less costly error to predict 2 than 3. We may set $L(1, 2) = 1/2$ and $L(1, 3) = 1$ accordingly. Then a distinct classifier will minimize risk.

13.6.1 Prior Probabilities

It is important to understand the role of the prior probabilities $\tilde{\pi} = (\pi_1, \ldots, \pi_m)$, since the optimal properties of the Bayes classifier depend on their correct identification. To see this, suppose we are developing a test for the presence of a type of infection. When we develop the test, we presumably have training data $(\mathbf{y}, \mathbf{X})$ with which to estimate conditional densities $f(\dot{x} \mid y = j)$. However, it would usually not be appropriate to use as estimates of $\tilde{\pi}$ the proportions of each class in the training data. We would hope that the prior probability of infection, say π_1, would be much less than $1/2$, and so for the purposes of efficient estimation, the proportion of the infected class in $\mathbf{y}$ should be chosen to be much higher than π_1.

For this reason, the choice of prior probabilities is often made independently of the training data. To take an extreme example, suppose the infection in question is nonexistent (small pox, for example). In this case, it would be appropriate to set $\pi_1 = 0$, in which case the Bayes classifier will predict noninfection for any observation $\dot{x}$.

When there is apparently no basis on which to choose prior probabilities a commonly used strategy is to select an *uninformative prior*, which weights each prediction equally. When the number of classes is finite, the logical choice would be the *uniform prior* $\pi_j = 1/m$. This is an example of the *principle of indifference*. However, in more complex example of Bayesian prediction, the question of what constitutes an uninformative prior can be a very deep one, and there may be a number of competing answers.

For our purposes, there are three choices:

1. We use prior knowledge to inform the choice of $\tilde{\pi}$.

2. We use a uniform prior as the indifferent choice.

3. We as estimates for $\tilde{\pi}$ the class frequencies observed in the training data

The third option is the easiest to take, since it is often the default choice of algorithms which build classifiers. Sometimes it will be the correct choice, but the fact that a choice is being made should always be kept in mind.

13.6.2 Naive Bayes Classifiers

Typically, the technical issue for Bayes classifiers is the estimation of the conditional density $f(\dot{x} \mid y = j)$ in (13.12). In principle, this may be done directly from the training data. However, there is scalability issue with respect to the number of features p. If we consider, for example, the p-dimensional multivariate normal distribution we note that the mean vector contains p parameters, while the covariance matrix contains $p + p(p-1)/2$ parameters (p variances and $p(p-1)/2$ covariances). This means the number of parameters to be estimated is of order $O(p^2)$, and so does not scale well with increasing numbers of predictors. A simple (ie 'naive') solution is to assume that the features are independent (even when evidence to the contrary exists), so that the conditional density is

$$f(\dot{x} \mid y = j) = \prod_{i=1}^{p} f_i(x_i \mid y = j).$$

Instead of estimating one multivariate density $f(\dot{x} \mid y = j)$, p univariate densities $f_i(x_i \mid y = j)$ are estimated, so that the total number of parameters to estimate is of order $O(p)$. In the normal case, that number is exactly $2p$, absent any further constraints.

13.7 K Nearest Neighbor (KNN) Classifiers and Regression

Assume there is a distance function d defined on the feature space $\mathcal{E}_{\mathbf{x}}$. We have data set $(\mathbf{y}, \mathbf{X})$. For any feature $\dot{x}$ and $K \geq 1$ define the neighborhood

$$\mathcal{N}_K(\dot{x}) = \{i : \text{rank of } d(\dot{x}_i, \dot{x}) \text{ no greater than } K \},$$

that is, $\mathcal{N}_K(\dot{x})$ consists of the indices of the K features nearest to $\dot{x}$ ($\dot{x}$ need not be in $\mathbf{X}$). Then for quantitative responses, the KNN predictor of $f(\dot{x})$ is

$$\hat{f}(\dot{x}) = \frac{1}{K} \sum_{i \in \mathcal{N}_K(\dot{x})} y_i,$$

or, the mean response in the neighborhood.

For the classification problem,

$$\hat{f}(\dot{x}) = j \text{ if } j \text{ is the most frequent class in } \mathcal{N}_K(\dot{x}),$$

where ties are resolved randomly.

In a typical application, Algorithms 12.1-12.2 may be used with the model spaces $\mathcal{F}_1, \ldots, \mathcal{F}_N$, where $\mathcal{F}_i$ is the model space for the KNN classifier with neighborhood size parameter K_i.

13.8 Linear and Quadratic Discriminant Analysis

Linear and quadratic discriminant analysis (LDA and QDA) are special cases of Bayes classifiers based on the normal distribution. Suppose, we have p features and m classes, and we wish to build a Bayes classifier with conditional distributions

$$f(\dot{x} \mid y = j) = \phi(\dot{x}; \boldsymbol{\mu}_j, \Sigma_j), \;\; j = 1, \ldots, m.$$

This requires an estimate of mean vectors and covariance matrices $\boldsymbol{\mu}_j, \Sigma_j$ for each class j. As discussed in Section 13.6.2 without further constraint the number of parameters to estimate becomes very large with increasing p. One way to control this is to assume (if justified) that the class covariance matrices are equal, so that $\Sigma_j = \Sigma$ for any j. Another method is to use a naive Bayes classifier (Section 13.6.2). In this case, the feature independence assumption forces each Σ_j to be a diagonal matrix, but they may still differ by class.

Once the conditional distributions are given the Bayes classifier is given directly by (13.12). It is generally simpler to apply a log transformation, in which case we have

$$\log(\phi(\dot{x}; \boldsymbol{\mu}_j, \Sigma_j)\pi_j) = -\frac{1}{2} Q_j(\dot{x}) - \frac{1}{2} \log(\det(\Sigma_j)) + \log(\pi_j) - \frac{p}{2} \log(2\pi), \tag{13.13}$$

where

$$Q_j(\dot{x}) = (\dot{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\dot{x} - \boldsymbol{\mu}_j).$$

At one level, all that is needed at this point is to calculate (13.13) for each class for any given observation $\dot{x}$, then take as the prediction that class with the largest value. If needed, the parameters can be estimated, as discussed in Section 13.8.1 below.

However, some insight can be gained by trying to refine the approach. First, we note that the procedure can be represented as a collection of functions $h_j(\dot{x})$, $j = 1, \ldots, m$, yielding prediction

$$\hat{y} = \text{argmax}_j h_j(\dot{x}). \tag{13.14}$$

This means these functions can be subject to a common strictly increasing transformation while yielding exactly the same predictions. This includes addition of a constant (positive or negative), or multiplication by a positive scalar. If we initially set $h_j(\dot{x})$ equal to (13.13), we might then note that there is a common term $-\frac{p}{2}\log(2\pi)$ (that it, it does not depend on j). We can therefore remove this term to get

$$h_j(\dot{x}) = -\frac{1}{2}Q_j(\dot{x}) - \frac{1}{2}\log(\det(\Sigma_j)) + \log(\pi_j).$$

Next, suppose we adopt a uniform prior $\pi_j = 1/m$ (Section 13.6.1). The term $\log(\pi_j)$ is now constant across classes, so it can be rmoved, yielding classifiers:

$$h'_j(\dot{x}) = -\frac{1}{2}Q_j(\dot{x}) - \frac{1}{2}\log(\det(\Sigma_j)). \tag{13.15}$$

Finally, suppose the covaniance matrices $\Sigma_j = \Sigma$ are constant. The term $-\frac{1}{2}\log(\det(\Sigma_j))$ may then be removed, but further simplification is possible. The quadratic term $Q_j(\dot{x})$ may also be decomposed:

$$\begin{aligned} Q_j(\dot{x}) &= (\dot{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\dot{x} - \boldsymbol{\mu}_j) \\ &= \dot{x}^T \Sigma_j^{-1} \dot{x} - 2(\dot{x})^T \Sigma_j^{-1} \boldsymbol{\mu}_j + \boldsymbol{\mu}_j^T \Sigma_j^{-1} \boldsymbol{\mu}_j. \end{aligned}$$

When $\Sigma_j = \Sigma$ the first term of this decomposition is constant across classes, so we have classifiers

$$h''_j(\dot{x}) = \dot{x}^T \Sigma^{-1} \boldsymbol{\mu}_j - \frac{1}{2}\boldsymbol{\mu}_j^T \Sigma^{-1} \boldsymbol{\mu}_j. \tag{13.16}$$

Note that the covariance matrix Σ still appears in (13.15), but only in terms which otherwise vary by class.

At this point we have the distinction between *linear* and *quadratic* discriminant analysis, specifically, whether or not the covariance matrices differ by class, which yield respective classifiers (in their most general form):

$$\begin{aligned} &LDA \qquad h_j(\dot{x}) = \dot{x}^T \Sigma^{-1} \boldsymbol{\mu}_j - \frac{1}{2}\boldsymbol{\mu}_j^T \Sigma^{-1} \boldsymbol{\mu}_j + \log(\pi_j) \\ &QDA \qquad h_j(\dot{x}) = -\frac{1}{2}(\dot{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\dot{x} - \boldsymbol{\mu}_j) - \frac{1}{2}\log(\det(\Sigma_j)) + \log(\pi_j). \end{aligned}$$

Recall the example in Figure 12.1. In general, classifiers can be defined geometrically by the boundaries they induce in the feature space. One advantantage of discriminant analysis is that this

boundary takes an analytical form. Suppose we have $m = 2$ classes. From (13.14) we can see that this boundary is given by the equation

$$h_1(\dot{x}) - h_2(\dot{x}) = 0.$$

By (13.17) it can be seen that for LDA this boundary is linear (boundary A, Figure 12.1) and for QDA it will be quadratic (boundary B, Figure 12.1).

Finally, the naive Bayes classifier (Section 13.6.2) is easily defined. For a multivariate normal distribution, independence is equivalent to zero covariance. Therefore a naive Bayes classifier is implemented simply by forcing each Σ_j to be a diagional matrix. This holds for both LDA and QDA. This method is known as *diagonal discriminant analysis* (DDA).

13.8.1 Estimation for LDA/QDA

The estimation problem is straightforward. We are given training data $(\mathbf{y}, \mathbf{X})$. The feature matrix is then partitioned by class, into $\mathbf{X}_1, \ldots, \mathbf{X}_m$. The class mean vector $\boldsymbol{\mu}_j$ is estimated by the sample means $\bar{\boldsymbol{\mu}}_j = (\bar{x}_1, \ldots, \bar{x}_p)$, where $\bar{x}_i$ is the sample mean of feature i using class j feature data $\mathbf{X}_j$.

Similarly, Σ_j is estimated by the sample covariance matrix using feature data $\mathbf{X}_j$:

$$\hat{\Sigma}_j = \frac{1}{n_j - 1} \begin{bmatrix} \sum_{i=1}^{n_j}(x_{i1} - \bar{x}_1)^2 & \cdots & \sum_{i=1}^{n_j}(x_{i1} - \bar{x}_1)(x_{ip} - \bar{x}_p) \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^{n_j}(x_{ip} - \bar{x}_p)(x_{i1} - \bar{x}_1) & \cdots & \sum_{i=1}^{n_j}(x_{ip} - \bar{x}_p)^2 \end{bmatrix}.$$

where n_j is the number of observations of class j, and the indices are defined relative to $\mathbf{X}_j$.

If we assume identical covariance matrices $\Sigma = \Sigma_j$, then the single estimate of Σ may be obtained by *pooling* the separate class estimates:

$$\hat{\Sigma}_{pooled} = \frac{1}{n - p} \sum_{j=1}^{p} (n_j - 1)\hat{\Sigma}_j,$$

(compare this procedure to the pooled t-test).

For DDA, the nondiagonal terms of $\hat{\Sigma}_j$ are set to zero, and the diagonal estimated as already described.

13.9 Logistic Regression

Consider a vector of responses $\mathbf{y}$ and a single predictor $\mathbf{x}$. So far, our regression models have assumed that the response has been normally distributed. We also noted that this assumption can be relaxed somewhat, with the inference methods for $\hat{\beta}_0, \hat{\beta}_1$, based on the t-distribution, remaining a reasonably accurate approximate.

Suppose. however, that are interested in predicting a binary outcomes. For example, we might track whether or not a clinic patient has a certain infection. Then the response would be, say, $y_i = 1$ if the patient is infected, and $y_i = 0$ otherwise. Thus, y_i is a Bernoulli random variable, and the normal assumption would not be appropriate.

Recall that when we develop a regression function

$$\hat{f}(x) = \hat{\beta}_0 + \hat{\beta}_1 x$$

we can think of $\hat{f}$ either as a prediction of a future response y give feature x, or as an estimate of the expected response $E[y]$. Similarly, for binary response, we may regard the problem as one of developing a function which estimates the expected value

$$P(A) = E[y_i] = g(x_i) \tag{13.17}$$

where we recognize y_i as the indicator function $I\{A\}$. In our example, $A = \{\text{patient is infected}\}$.

The most common approach to this problem is the *logistic regression model*. We retain much of the structure of linear regression. For example, we have the usual prediction matrix $\mathbf{X}$ and linear coefficients

$$\eta = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p.$$

However, η is not related to the response via the linear regression model $y = \eta + \epsilon$. Noting that an estimate of $E[y_i]$ should sensibly be forced into the unit interval $[0, 1]$, we select g in (13.17) which does this, which then relates η to the response:

$$E[y_i] = g(\eta). \tag{13.18}$$

The final choice is of g. Several are proposed in the literature, but the most commonly used is the *logistic function*

$$g(\eta) = \frac{e^\eta}{1 + e^\eta} = \frac{1}{1 + e^{-\eta}} \in (0, 1) \tag{13.19}$$

This completely specifies the model:

$$y_i \sim bern(g(\eta_i)), \text{ where } \eta_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}.$$

13.9.1 The Odds Ratio in Logistic Regression

One advantage of the logistic function (13.19) is that it gives a very convenient method of calculating the odds ratio of the defining event $y_i = 1$ between two predictor values. Given response/predictor pair $(y, \dot{x})$ we have estimate

$$\begin{aligned} P(y = 1) &\approx \frac{1}{1 + e^{-\hat{\boldsymbol{\beta}}^T \dot{x}}}, \text{ and} \\ Odds(y = 1) &\approx e^{\hat{\boldsymbol{\beta}}^T \dot{x}}. \end{aligned}$$

Given two predictor observations $\dot{x}, \dot{x}'$ the odds ratio between them is therefore estimated by

$$OR(y = 1; \dot{x}, \dot{x}') \approx e^{\hat{\boldsymbol{\beta}}^T (\dot{x} - \dot{x}')}.$$

13.9.2 Likelihood Method for Logistic Regression

The density function for a Bernoulli random variable $Y \sim bern(\pi)$ can be written

$$f(y) = \pi^y (1 - \pi)^{1-y}.$$

Then set expected values of y_i to be

$$\pi_i = g(\eta_i).$$

Note that π_i is ultimately a function of the regression coefficients β_i. If we assume the responses are independent (as is commonly done), the likelihood function is, after some algebra,

$$l(\hat{\boldsymbol{\beta}}; \tilde{y}) = \prod_{i=1}^{n} \pi_i^{y_i} (1 - \pi_i)^{1-y_i}, \tag{13.20}$$

and the log-likelihhood function is

$$L(\hat{\boldsymbol{\beta}}; \tilde{y}) = \sum_{i=1}^{n} y_i \log(\pi_i) + (1 - y_i) \log(1 - \pi_i).$$

We can use the methods outlined in Chapter 7 to obtain MLEs and standard errors for $\hat{\boldsymbol{\beta}}$. This is what most software does. However, there is no closed form solution to the optimization problem, so numerical algorithms are used.

We would also like to develop a goodness of fit measure, since this is an important tool for model selection. Although the model can be considered a form of classification, it yields a quantitative estimate in the form a probability, rather than a predicted class, so classification error CE is not a natural choice.

We then note that in linear regression, the goodness of fit measure is based on the criterion which is optimized in order to fit the model, which is the MSE. In logistic regression, it is the likelihood which is optimized.

In logistic regression, the fitted values are $\hat{\pi}_i$, the estimates of $E[y_i]$. Next, recall from the theory of linear regression the notion of a full and reduced model.

The Null Model

Recall that the simplest regression model is the one for which all coefficients are zero except the intercept:

$$y_i = \beta_0 + \epsilon_i, \tag{13.21}$$

that is, the predictors play no role. In this case, the least squares (and the maximum liklihood) estimate is $\hat{\beta}_0 = \bar{y}$.

The same logic applies to logistic regression. If the linear prediction term is simple $\eta_i = \beta_0$, it is easily shown that the maximum likehood estimate of the fitted values is simply

$$\hat{\pi}_i = \hat{\pi}_{null} = \frac{1}{n} \sum_{i=1}^{n} y_i, \text{ for all } i,$$

which is the observed probability that $y_i = 1$ (the estimate $\hat{\beta}_0$ is whatever value uniquely achieves this). The *null likelihood* is then

$$l_{null} = \hat{\pi}_{null}^{n_1} (1 - \hat{\pi}_{null})^{n-n_1}$$

where n_1 is the number of responses equal to one.

The Fitted Likelihood

Once the MLEs $\hat{\beta}_i$ are calculated, they can be substituted back into the linear predictor terms to yield

$$\hat{\eta}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}$$

and

$$\hat{\pi}_i = g(\hat{\eta}_i).$$

Subtituting into (13.20) yields the *fitted likelihood*

$$l_{fit} = l(\hat{\boldsymbol{\beta}}; \tilde{y}).$$

The Saturated Likelihood

Recall the saturated model of Section 7.1. If we had a 'perfect' model, then we would have enough predictors to force $y_i = \hat{y}_i$ for all i. As can be seen from (13.20) the likelihood would be equal to one in this case, yielding saturated likelihood:

$$l_{sat} = 1.$$

At this point recall the ANOVA structure in linear regression (Section 2.2). The total sum of squares decomposes as

$$SSTO = SSR + SSE.$$

While SSE is a goodness of fit measure for the regression model, it is also important to remember that SSTO is also interpretable as a special case of SSE for the null model (13.21), and SSR is reduction in SSE when predictors are added to the null model. We also have coefficient of determination

$$R^2 = \frac{SSTO - SSE}{SSTO} = \frac{SSR}{SSTO}. \tag{13.22}$$

Following Section 7.1 we may define *model deviance*,

$$D_{model} = -\left[2\log(l_{fit}) - 2\log(l_{sat})\right] = -2\log(l_{fit}),$$

and *null deviance*

$$D_{null} = -\left[2\log(l_{null}) - 2\log(l_{sat})\right] = -2\log(l_{null}).$$

These quantities serve as a type of ANOVA decomposition, with analogous relationships

$$\begin{aligned} D_{model} &\iff SSE, \\ D_{null} &\iff SSTO, \text{ and} \\ D_{null} - D_{model} &\iff SSR. \end{aligned}$$

By Wilk's theorem (7.4), under the null hypothesis

$$H_o : E[y_i] = \pi, \ i = 1, \ldots, n$$

that the null model is correct we have

$$D_{null} - D_{model} \sim \chi^2_p$$

approximately, where p is the number of predictors (in addition to the intercept).

In addition, it is common to define a *pseudo-R^2*, of which several forms exist. By direct analogy to linear regression we have

$$R^2_L = \frac{D_{null} - D_{model}}{D_{null}},$$

known as the *likelihood ratio* R^2 (compare to (13.22)). It is known that R^2_L is not in a monotonic relationship with the odds ratio (Section 13.9.1). The *Cox-Snell* R^2 is an alternative pseudo-R^2 defined by

$$R^2_{CS} = 1 - \left(\frac{l_{null}}{l_{model}}\right)^{2/n}.$$

This is a more natural choice for logistic regression based on maximum likelihood estimation. However, the maximum value of R^2_{CS} is

$$R^2_{CS} \leq 1 - (l_{null})^{2/n} < 1,$$

since R^2_{CS} would be maximized by the saturated model. For this reason, the *Nagelkerke* pseudo-R^2 is sometimes used, which is simply R^2_{CS} normalized to attain a maximum of 1:

$$R^2_N = \frac{R^2_{CS}}{1 - (l_{null})^{2/n}}.$$

13.10 Classification and the Receiver Operator Characteristic (ROC) Curve

We already introduced in Chapter 13 a probabilistic model for the evaluation of a classifier, in the context of diagnostic testing. This was based on an application of Baye's theorem to the following events on a probability space:

$$\begin{aligned} O_+ &= \{ \text{ positive outcome } \} \\ O_- &= \{ \text{ negative outcome } \} \\ T_+ &= \{ \text{ positive test outcome } \} \\ T_- &= \{ \text{ negative test outcome } \}. \end{aligned}$$

We defined sensitivity and specificity as the following quantities:

$$\begin{aligned} sens &= P(T_+ \mid O_+) \\ spec &= P(T_- \mid O_-), \end{aligned}$$

and we may also define the *false positive rate* and *false negative rate* as

$$\begin{aligned} fpr &= P(T_+ \mid O_-) = 1 - spec \\ fnr &= P(T_- \mid O_+) = 1 - sens. \end{aligned}$$

These quantities are relevant in the evaluation phase of the development of a classifier. The ultimate goal is to maximize the positive predictive value (PPV) and negative predictive value (NPV), defined as

$$\begin{aligned} PPV &= P(O_+ \mid T_+) \\ NPV &= P(O_- \mid T_-), \end{aligned}$$

but to do so we need to test the classifier using subjects with known outcomes O_+ and O_-, which gives *sens* and *spec*. We also need the prevalence of the outcome

$$prev = P(O_+),$$

with which Baye's theorem leads to

$$\begin{aligned} PPV &= P(O_+ \mid T_+) \\ &= \frac{P(T_+ \mid O_+)P(O_+)}{P(T_+ \mid O_+)P(O_+) + P(T_+ \mid O_-)P(O_-)} \\ &= \frac{sens \times prev}{sens \times prev + (1 - spec) \times (1 - prev)} \end{aligned}$$

and

$$\begin{aligned} NPV &= P(O_- \mid T_-) \\ &= \frac{P(T_- \mid O_-)P(O_-)}{P(T_- \mid O_-)P(O_-) + P(T_- \mid O_+)P(O_+)} \\ &= \frac{spec \times (1 - prev)}{spec \times (1 - prev) + (1 - sens) \times prev}. \end{aligned}$$

13.10.1 Classifiers Based on a Numerical Risk Score

The preceding section summarizes a probabilistic classification model for which the classifier can be reduced to two outcomes T_+ and T_-. Of course, classifiers are often based on a numerical score. We can adopt the convention that higher scores can be interpreted as evidence in favor of a positive outcome O_+ (if needed, reverse the score by multiplying by -1). In this way, the numerical classifier can be interpreted as a *risk score*, with high risk implying greater probability (risk) of a positive outcome O_+.

To fix ideas, consider the `Melanoma` data set included in the `MASS` package:

```
> library(MASS)
> help(Melanoma)

  Survival from Malignant Melanoma

Description

The Melanoma data frame has data on 205 patients in
Denmark with malignant melanoma.
```

```
Usage

Melanoma
Format

This data frame contains the following columns:

time
survival time in days, possibly censored.

status
1 died from melanoma, 2 alive, 3 dead from other causes.

sex
1 = male, 0 = female.

age
age in years.

year
of operation.

thickness
tumour thickness in mm.

ulcer
1 = presence, 0 = absence.

Source

P. K. Andersen, O. Borgan, R. D. Gill and
N. Keiding (1993) Statistical Models based on Counting
Processes. Springer.
```

We will investigate the possibility of using `thickness` (tumor thickness in mm) to predict death from melanoma. We have outcome `status`, which classifies the patient as dead from melanoma (`status = 1`); alive (`status = 2`); or dead from other causes (tttstatus = 3). We may remove from the analysis patients who died from other causes, leaving outcomes

$$\begin{aligned} O_+ &= \{\text{patient died from melanoma}\} \\ O_- &= \{\text{patient is still alive}\}. \end{aligned}$$

In practice, this type of analysis would take into account the observation times of the patients, which may vary considerably. For example, a patient with outcome O_- may have only been observed for

a short period of time, so that that negative outcome would be more difficult to interpret than an negative outcome which follows a longer observation period. With that caveat, we will accept survival as the outcome.

We have a quick first look at the data:

```
> names(Melanoma)
[1] "time"      "status"    "sex"       "age"       "year"
 "thickness"    "ulcer"
> Melanoma[1:3,]
  time status sex age year thickness ulcer
1   10      3   1  76 1972      6.76     1
2   30      3   1  56 1968      0.65     0
3   35      2   1  41 1977      1.34     0
> is.factor(Melanoma$status)
[1] FALSE
>
```

Note that `status` is not a factor variable. So, to subset the data we use the command:

```
> Melanoma2 = Melanoma[Melanoma$status < 3,]
> dim(Melanoma2)
[1] 191   7
>
```

and use data frame `Melanoma2`. There are $n = 191$ subjects remaining.

Next, look at boxplots of the variable `thickness` by outcome group (Figure 13.2):

```
> par(mfrow=c(1,1), mar=c(3,5,3,3), cex=1.1)
> boxplot(thickness ~ status, data = Melanoma2,
     names = c("Died", "Alive"), ylab="Tumor Thickness in mm.")
> for (i in 1:10) {lines(c(0,3), rep(i,2), col=4)}
```

We have superimposed lines (in blue) at `thickness` levels $1, 2, \ldots, 10$. Clearly, death outcomes are associated with higher values of `thickness`, which can therefore be used as a risk score for melanoma mortatility (higher values of `thickness` mean greater mortality risk). Suppose we select a *risk score threshold* T, possibly one of the blue lines. We may then define a positive test outcome as

$$T_+ = \{thickness \geq T\}. \tag{13.23}$$

This allows us to apply a classifier in an intuitive way, in the sense that if T_+ occurs we predict O_+, and if $T_- = T_+^c$ occurs we predict O_-.

Of course, this leaves open the problem of selecting T. If there were no overlap (that is, if the smallest risk score among the O_+ group was larger than the largest risk score among the O_- group) then the selection of T would be straightforward. If there was some T that is larger than all risk scores in the O_- group and smaller than all risk scores in the O_+ group, we would would use that threshold to define a positive test according to (13.23), which would yield $sens = spec = 1$ that is, perfect classification (at least for this sample).

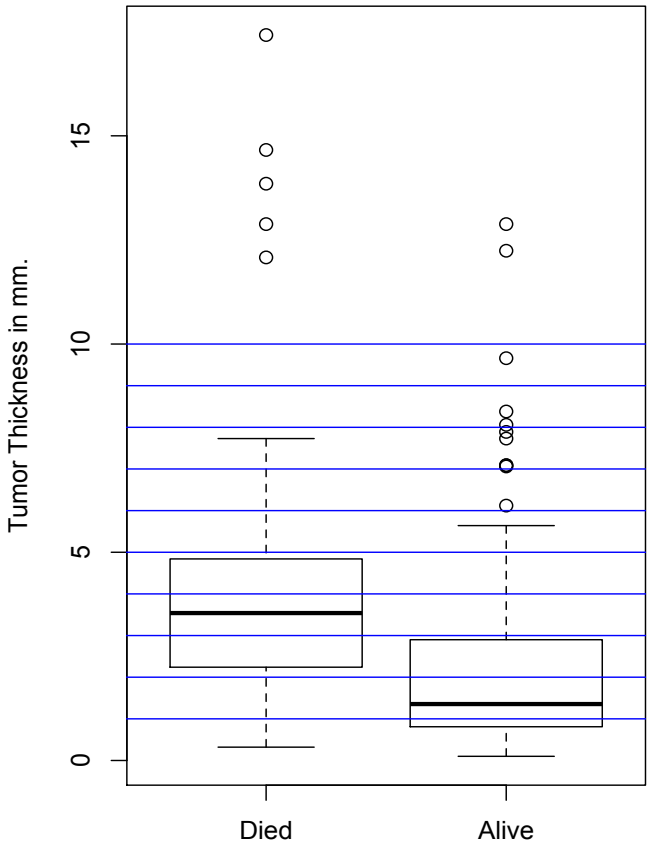

Figure 13.2: Boxplot of melanoma *thickness* variable (tumor thickness in mm) by survival outcome group.

Of course, we don't usually expect this ideal. Suppose we consider the blue lines in Figure 13.2 as possible values for the threshold T used to define the positive test outcome T_+ in (13.23). Clearly, for each value of T we will have false positives and false negatives, as long as within each group there are risk scores on both sides of the threshold.

It is instructive, however, to consider the limiting case. If $T = 0$ (all risk scores are above 0), then the test outcome will be positive for *all* subjects in *both* groups. Since mortality is (correctly) predicted for all subjects in O_+, we have $sens = 1$. At the same time, mortality is (incorrectly) predicted for all subjects in O_-, so $spec = 0$. This is clearly not a satisfactory predictor. If $T = 100$ (that is, a value larger than all observed risk scores) we (incorrectly) predict survival for all subjects in O_+, so that $sens = 0$. We also (correctly) predict survival for all subjects in O_-, so that $spec = 1$.

Clearly, we must find a balance between *sens* and *spec*. As T increases, *sens* decreases and *spec* increases. At this point, we can write an `R` function that calculates *sens* and *spec* for a given threshold, for a given data set. The function will have to input three things, namely, the threshold

T, risk score *score* and the outcome groups gr. The variable gr will be a 0-1 numerical vector, with 1 corresponding to high risk. We assume *score* and gr are paired. Subjects with $score \geq T$ are assigned positive test outcomes.

To estimate *sens* and *spec* from the data, we can use the following calculation:

$$\begin{aligned} sens &= \frac{P(O_+ \cap T_+)}{P(O_+)} = \frac{\text{Num subjects for which } score \geq T \text{ and } gr = 1}{\text{Num subjects for which } gr = 1} \\ spec &= \frac{P(O_- \cap T_-)}{P(O_-)} = \frac{\text{Num subjects for which } score < T \text{ and } gr = 0}{\text{Num subjects for which } gr = 0}. \end{aligned}$$

We therefore write the function:

```
> diag.thresh = function(thresh, score, gr) {
+  sens= sum( (score >= thresh) & (gr == 1) )/sum(gr == 1)
+  spec= sum( (score < thresh) & (gr == 0) )/sum(gr == 0)
+  ans = c(sens, spec)
+  names(ans) = c("Sensitivity", "Specificity")
+  return(ans)
+ }
>
```

We can create our data variables for input

```
> gr = 1*(Melanoma2$status == 1)
> score = Melanoma2$thickness
> gr = gr[sort.list(score)]
> score = score[sort.list(score)]
```

Note that we have sorted the paired data using the `sort.list()` function. We can, for example, get the sensitivity associated with a threshold of $T = 5$:

```
> diag.thresh(5, score, gr)
Sensitivity Specificity
  0.2456140   0.8955224
>
```

While the specificity is quite good ($spec = 0.8955224$) the sensitivity would be, by most standards, too low ($sens = 0.2456140$), and we would probably want to use a lower threshold for an actual application.

To see how the specificity and sensitivity vary with threshold T, we can create a loop to calculate a range of values for T, and create a simple table.

```
> diag.tab = NULL
>  for (i in 1:10) {diag.tab =
                rbind(diag.tab,diag.thresh(i, score, gr))
+ }
> rownames(diag.tab) = paste("Threshold",1:10)
```

```
> colnames(diag.tab) = c("Sensitivity", "Specificity")
> diag.tab
            Sensitivity Specificity
Threshold 1    0.8947368   0.3507463
Threshold 2    0.7719298   0.6641791
Threshold 3    0.5964912   0.7611940
Threshold 4    0.3859649   0.8656716
Threshold 5    0.2456140   0.8955224
Threshold 6    0.1578947   0.9179104
Threshold 7    0.1403509   0.9253731
Threshold 8    0.0877193   0.9626866
Threshold 9    0.0877193   0.9776119
Threshold 10   0.0877193   0.9850746
>
```

A value of T in the range 2 to 3 would seem to offer a better balance of false positive and false negative rates.

13.10.2 ROC Curves

Of course, this type of analysis can be much more refined. First of all, we can input as threshold T all observed value of the risk score, obtaining much greater resolution than the previous table. To do this, we could use a `for` loop, but it's good to remember at this point that `R` permits vectorized operation. This would seem to suggest that if in the function call `diag.thresh(i, score, gr)` we substitute `score` for i, we would get the $sens, spec$ values for all observed values of *score* in a single object. However, if we try this we get:

```
> diag.thresh(score, score, gr)
Sensitivity Specificity
          1           0
>
```

which is not what we wish. The problem lies in the fact that the other inputs are also vectors, leading to ambiguity. The problem may be fixed by using the `Vectorize()` function, which modifies an existing function by designating one or more of it's inputs as the *vectorized* input as an option. The original function is evaluated for each element of the vectorized input. A new function is created in this way:

```
> help(Vectorize)
> diag.thresh.vect = Vectorize(diag.thresh, "thresh")
> temp = diag.thresh.vect(score,score,gr)
> dim(temp)
[1]   2 191
> sens = temp[1,]
> spec = temp[2,]
```

A new function `diag.thresh.vect()` has been created, which evaluates `diag.thresh()` separately for each element of the vector used as the first argument. The results are stored as a 191×2 matrix, each column containing the values of *sens*, *spec* for each element of `score`.

At this point we are ready to plot an *ROC curve*, which is simply a plot of sensitivity (or true positive rate) against 1-specificity (or false positive rate) ('ROC' is an acronym for *receiver operating characteristic*). The script used to draw the plot is given below (Figure 13.3).

```
> auc = roc.area(class, gr)
> pv = wilcox.test(gr ~ class)$p.value
> par(mfrow=c(1,1), cex=1.1, oma = c(1,2,1,1))
> plot(1-spec, sens, xlab="false positive rate (1 - specificity)",
         ylab="true positive rate (sensitivity)", type = "s")
> title("ROC curve for prediction of melanoma
        survival \n based on tumor thickness")
> lines(c(0,0),c(0,1),col=3)
> lines(c(0,1),c(1,1), col=3)
> lines(c(0,1), c(0,1))
> text(.7,.1, paste("AUC = ",signif(auc,3),",
         P = ", signif(pv,3),sep=""))
```

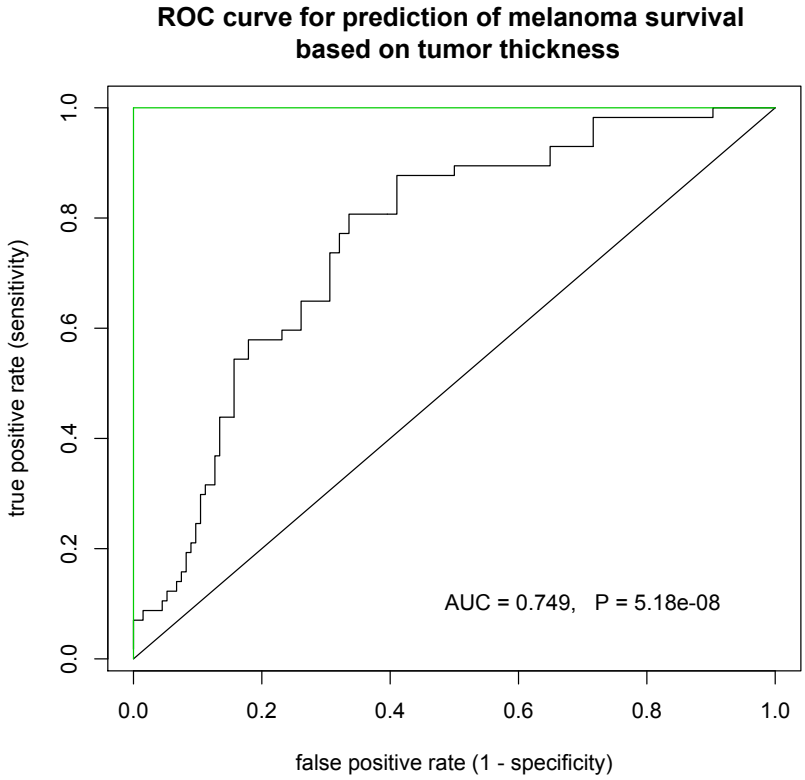

Figure 13.3: ROC curve for prediction of melanoma cancer survival based on melanoma *thickness* variable (tumor thickness in mm). The AUC is given, as well as the p-value for a WIlcoxon rank sum test for group homogeneity of risk score distributions. The green lines represent a "perfect" classifier, with sensitivity and specificity both equal to one. The diagonal identity line represents a noninformative risk score of $AUC = 0.5$.

First note the option `type = "s"` in the `plot()` function, which produces a step function type plot, which is appropriate for an ROC curve. Also, the control character `‘‘\n"` may be used in the plot title to force a line break. In addition, an identity reference line has also been added to the plot, as well as green lines joining the points (0,0), (0,1) and (1,1). The plot also gives two quantities, AUC as well as a p-value, which we now explain.

First, recall the "perfect" classifier discussed above, with sensitivity and specificity both equal to one. In this case, the ROC curve would coincide with the green lines of Figure 13.3. A highly accurate risk score would produce an ROC curve close to the green lines in some sense.

We next explain AUC. This is simply an acronym for *area under curve*. That is, AUC is defined as the area under the ROC curve. It may be shown that AUC is equal to the probability that a randomly chosen positive subject has a higher risk score than a randomly chosen negative

subject. This can be given directly from the data:

$$AUC = \frac{\sum_{i\in\text{-ve}} \sum_{j\in+\text{ve}} I\{score_j > score_i\} + 0.5 \times I\{score_j = score_i\}}{n_- \times n_+} \tag{13.24}$$

where n_-, n_+ are the number of negative and positive outcome subjects. Note that ties are assumed to be resolved randomly, hence the presence of the 0.5 factor in the numerator of (13.24). A function which calculates AUC is given below, and was used to calculate the value of AUC shown in figure Figure 13.3. This function is not, but could be, vectorized, as for the `diag.thresh.vect()` function given above.

```
> roc.area<-function(x,y) {
+
+  y0 = y[x==0]
+  y1 = y[x==1]
+
+  count<-0
+  for (i in 1:length(y0))
       {count = count+sum(y1 > y0[i]) + 0.5*sum(y1 == y0[i])}
+  ans = count/(length(y0)*length(y1))
+  return(ans)
+ }
>
```

Suppose, in contrast to the perfect classifier indicated by the green line in Figure 13.3, that the risk score actually contains no information about the outcome. In this case, a randomly selected positive subject is equally likely to have a higher or lower risk score than a randomly selected negative subject. In this case we would expect $AUC = 0.5$, and the ROC curve would therefore lie on the indentity. For this reason, the identity line is often included in an ROC curve graphic, and the degree to which the ROC curve lies above the identity gives a direct assessment of the predictive value of the risk score (the green lines are usually not given). What would you conclude if the ROC curve lay significantly *below* the identity?

Finally, we explain the p-value. It may be shown mathematically that the AUC is equivalent to the Wilcoxon rank sum statistic for a comparison of the risk score between the two outcome groups. This means the Wilcoxon rank sum test is interpretable as a test against the null hypothesis $H_o : AUC = 0.5$. For this reason the p-value may be used to confirm that the risk score is significantly predictive of the outcome in a formal statistical sense.

Chapter 14

Unsupervised Learning

The distinction between supervised and unsupervised learning was discussed in Chapter 12. As in that chapter we have an $n \times p$ matrix of data

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ip} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_i \\ \vdots \\ \dot{x}_n \end{bmatrix} = [\mathbf{x}_1 \cdots \mathbf{x}_p] \,. \tag{14.1}$$

The columns $\mathbf{x}_j$ of $\mathbf{X}$ represent p features, or types of information. The rows $\dot{x}_i$ of $\mathbf{X}$ represent n observations associated with, for example, individual subjects in a study. Then $\dot{x}_i$ contains the specific value of each feature for subject i. Note that we do not have a response variable $\mathbf{y}$ as would be needed for supervised learning. The object is to partition the subjects $\{1, \ldots, n\}$ into clusters $A_1, \ldots, A_m$, each subject belonging to exactly one of the m clusters. Sometimes, the features may be clustered, in which case the methodology is the same. It will be important to review Section 12.3 on distances, since many unsupervised learning algorithms are based on an $n \times n$ distance matrix D, in which element d_{ij} is a distance between observations i and j.

14.1 Hierarchical Clustering

.

We are given an $n \times n$ pairwise distance matrix D for n observations. We can used D to define a *cluster distance*, that is, a way of measuring the distance between two clusters of observations A, B, or alternatively, clusters of indices from $\{1, \ldots, n\}$. Three commonly used methods are given below:

- **Single link (connected)**. Distance between nearest observations.

$$D(A, B) \;=\; \min\{d_{ij} : i \in A, j \in B\}$$

- **Compact**. Distance between furthest objects.

$$D(A, B) \;=\; \max\{d_{ij} : i \in A, j \in B\}$$

- **Average**. Average distance.

$$D(A,B) = \frac{1}{|A||B|}\sum_{i\in A}\sum_{j\in B} d_{ij}$$

Note that if A and B each consist of a single label i and j then $D(A,B) = d_{ij}$ for each of the above methods.

Hierarchical clustering proceeds using the following steps:

1. Define a cluster for each observation.

2. Form a cluster from the two observations with the smallest pairwise distance.

3. There are now $n-1$ 'clusters', one with two observations, $n-2$ with one observation each.

4. Successively join the two clusters with the shortest distance D between them.

The resulting cluster is usually represented by a **dendogram**. This is a tree in which terminal nodes represent the observations and the remaining nodes represent the cluster consolidations. Usually, the vertical distance corresponds to the actual cluster distances. A horizontal cross-section of a dendogram induces a partition.

Example 14.1. Single link clustering applied to the distance matrix below results in the dendogram shown in Figure 14.1.

```
Distance matrix:

     [,1] [,2] [,3] [,4] [,5]
[1,]    0    1   10   10   10
[2,]    1    0   10   10   10
[3,]   10   10    0    5    5
[4,]   10   10    5    0    5
[5,]   10   10    5    5    0
```

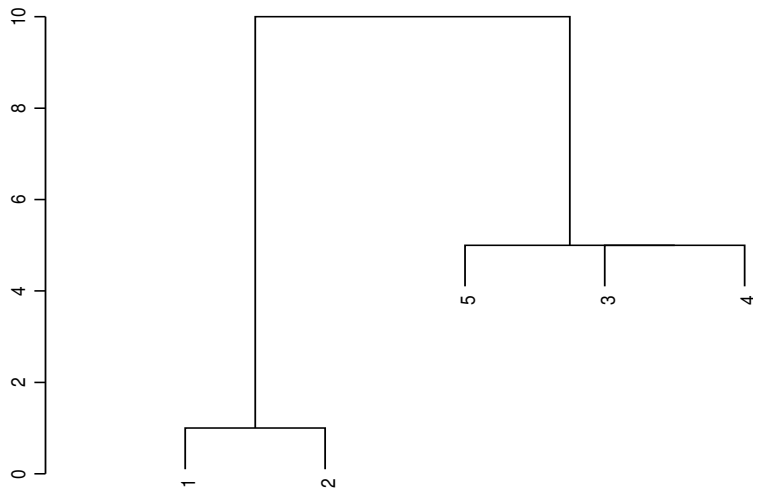

Figure 14.1: Dendogram for Example 14.1

14.2 K-Means Cluster Analysis

K-means clustering is a method in which a fixed number K of clusters is specified, and an attempt is made to find the partition of size K which minimizes some objective function. (We say 'attempt' because many of these algorithms are heuristic).

- We need to define a **centroid** $g(A) = (g_1(A), \ldots, g_p(A))$ of a cluster A of features. This may be the component-wise average of the features, but other alternatives are sometimes used.
- In the **sum of squares method** the objective function for partition $\tilde{A} = (A_1, \ldots, A_K)$ is the *within cluster sum of squares*

$$SS_{within} = \sum_{i=1}^{K} \sum_{j \in A_i} d(\dot{x}_j, g(A_i))^2$$

  where d is a distance function on the space of features.

As in linear regression models and ANOVA we can define a *total sum of squares*

$$SS_{total} = \sum_{i=1}^{n} d(\dot{x}_i, g(A_0))^2$$

where $g(A_0)$ is the centroid of the entire data set. By analogy, we can define a quantity similar to the coefficient of determination:

$$R^2 = 1 - \frac{SS_{within}}{SS_{total}}.$$

Values of R^2 close to one imply that most of the total variation is explainable by the clustering.

14.3 Principal Components Analysis

Principle components analysis is primarily a dimension reduction technique. Suppose we are given the $n \times p$ feature matrix defined in Equation (14.1). There may be considerable advantage to representing the feature space with fewer dimensions than p.

First, suppose A is a $p \times p$ matrix. We may then transform $\mathbf{X}$ through matrix multiplication

$$\mathbf{Y} = \mathbf{X}A,$$

so that $\mathbf{Y}$ is another $n \times p$ matrix, and can be interpreted in much the same way. The column vectors of $\mathbf{Y}$, $\mathbf{y}_1, \ldots \mathbf{y}_p$ are linear combinations of the original feature vectors in $\mathbf{X}$, with coefficients given by matrix A:

$$\begin{aligned}
\mathbf{y}_1 &= a_{11}\mathbf{x}_1 + \ldots + a_{p1}\mathbf{x}_p \\
&\vdots \\
\mathbf{y}_i &= a_{1i}\mathbf{x}_1 + \ldots + a_{pi}\mathbf{x}_p \\
&\vdots \\
\mathbf{y}_p &= a_{1p}\mathbf{x}_1 + \ldots + a_{pp}\mathbf{x}_p
\end{aligned}$$

The principal components $\mathbf{y}_1, \ldots \mathbf{y}_p$ are constructed using the following steps.

1. Normalize each column of $\mathbf{X}$ to have zero mean and unit variance (this step might be omitted for specific reasons).

2. To create the first principal component $\mathbf{y}_1$, determine coefficients $a_{11}, \ldots, a_{p1}$ which maximize tha variance of $\mathbf{y}_1$ subject to constraint

$$a_{11}^2 + \ldots + a_{p1}^2 = 1.$$

3. Successively create the remaining principal components in order. Create the ith principal component $\mathbf{y}_i$ by determining the coefficients $a_{1i}, \ldots, a_{pi}$ which maximize tha variance of $\mathbf{y}_i$ subject to constraint

$$a_{1i}^2 + \ldots + a_{pi}^2 = 1,$$

such that $\mathbf{y}_i$ is orthogonal to previous principal components $\mathbf{y}_1, \ldots, \mathbf{y}_{i-1}$.

If dimension reduction is feasible, then the first few principal components will have significantly greater variance than the remaining ones, so that the data set $\mathbf{X}$ can be approximately represented by those.

Chapter 15

Score Based Model Selection

So far, model selection has been based on goodness of fit measures such as SSE for least squares regression or deviance for likelihood based inference. These quantities are used to compare full and reduced nested models, or to estimate MSE_{test} for alternative model families using cross-validation.

Such methods are necessary because goodness of fit scores tend to reward model complexity. When comparing full and reduced nested models, the SSE is always smaller, and the likelihood, and therefore the model deviance, is always larger, for the full model. Even when models are not nested the trend is the same.

To some extent these effects are controlled by alternative scores. We have, for example,

$$MSE = \frac{SSE}{n-q},$$

where q is the number of parameters. With addition of a new predictor, both the numerator and denominator decrease, so MSE does not necessarily reward increasing complexity. However, as discussed in Section 12.5 if in a linear regression model the number of predictors $q = n$, the sample size, then $SSE = 0$ and $MSE = 0$. Furthermore, this bias towards zero can be observed well before that point is reached. The same is true, for the same reason, for the adjusted R^2_{adj} (Section 3.2.2)

$$R^2_{adj} = 1 - \frac{SSE/(n-p-1)}{SSTO/(n-1)},$$

for a regression model with p predictors.

In this chapter we consider alternative scores. Suppose, for example, that instead of minimizing SSE or model deviance D_{model}, we minimize

$$\Lambda = SSE + \lambda(\theta) \text{ or } \Lambda = D_{model} + \lambda(\theta) \tag{15.1}$$

where $\lambda(\theta)$ is a *complexity penalty* that depends on a parameter θ which represents the model. Possibly, $\lambda(\theta)$ is an increasing function of the number of model parameters q, so that the tendency of model complexity to force SSE to zero will be balanced by the complexity penalty.

There is, in fact, a rich theory behind such score based model selection techniques, and a number of widely used alternatives exist. Scores similar to Λ defined in (15.1) have been derived based on specific mathematical principles. Perhaps the two most widely used are the *Akaike information criterion* (AIC) and the *Bayesian information criterion* (BIC) (the BIC is also known

as the *Schwarz information criterion* (SIC)). The most general definition is given in terms of the likelihood function $l(\hat{\theta}_{MLE})$:

$$\begin{aligned} AIC &= -2\log(l(\hat{\theta}_{MLE})) + 2q, \\ BIC &= -2\log(l(\hat{\theta}_{MLE})) + \log(n)q, \end{aligned} \tag{15.2}$$

where n is the number of observations, and q is the number of model parameters (for linear regression, q is the number of predictors plus one for the intercept).

15.1 AIC and BIC for Multiple Linear Regression

There is a technical issue regarding the application of maximum likelihood estimation to linear regression, which relates to the term $-2\log(l(\hat{\theta}_{MLE}))$ appearing in (15.2). Following equation (7.1) of Example 7.1, for linear regression we have

$$-2\log(l(\hat{\theta}_{MLE})) = \frac{1}{\sigma^2}SSE + C, \tag{15.3}$$

where C is a constant which does not depend on unknown parameters, and may be set to zero for convenience. However, this was obtained assuming that σ^2 was fixed, or equivalently, did not need to be estimated using the data. We could also include σ^2 in the maximum likelihood calculation, in which case we would have

$$-2\log(l(\hat{\theta}_{MLE})) = n\log(SSE/n) + C, \tag{15.4}$$

where C is the same type of constant, and may be set to zero. The coefficient estimates $\hat{\boldsymbol{\beta}}$ are the same in both cases (that is, the least squares estimates), and in both cases σ^2 can be estimated in whatever manner is most appropriate. The distinction lies in the manner in which different models are compared.

15.1.1 Model Selection Algorithms Based on Predictor Subsets

Suppose we have a total of p predictors, which gives full (or *complete*) model

$$\mathbf{y} = \beta_0 + \beta_1\mathbf{x}_1 + \ldots + \beta_p\mathbf{x}_b + \boldsymbol{\epsilon}.$$

The problem is to decide which predictors to retain, assuming that all models contain the intercept. A model M can then be defined as a subset of indices $\{1, \ldots, p\}$ indicating the retained predictors. Model $M = \{1, 3\}$ then denotes

$$\mathbf{y} = \beta_0 + \beta_1\mathbf{x}_1 + \beta_3\mathbf{x}_3 + \boldsymbol{\epsilon}.$$

In principle, we can calculate

$$AIC_M = \frac{1}{\hat{\sigma}^2}SSE_M + 2k_M \tag{15.5}$$

for any model M, where SSE_M is the SSE for that model, k_M is the number of parameters in the model. We also need an estimate $\hat{\sigma}^2$ of σ^2 which is model independent. This is usually estimated using the MSE of the full model.

For *all subsets* model selection, the value AIC_M is calculated for all models, or equivalently, for all subjets M of index set $\{1, \ldots, p\}$. The model selected is the one that minimized AIC_M. Note that this includes the empty set, representing null model

$$\mathbf{y} = \beta_0 + \boldsymbol{\epsilon}.$$

If the null model is selected, the inference is that none of the predictors is significantly related to the response.

This seemingly straightforward method suffers from a serious drawback. Using the product rule of combinatorics, it can be seen that the total number of models is 2^p. We choose a model by deciding independently for each predictor to include or not include it. This represents p decisions of two outcomes. Note that $2^{10} = 1024$, $2^{15} = 32,768$, $2^{20} = 1,048,576$, and so on. Eventually, for large numbers of predictors, all subsets model selection will not be computationally feasible.

A reasonable approach is to order the predictors in decreasing order of importance. Assuming there is some basis on which to do this, we let M_0 be the null model, and let M_k be the model including the k most important predictors. The selected model is then simply

$$M^* = \text{argmin}_{M_k : k=0,\ldots,p} \, AIC_{M_k},$$

the model in the sequence with the minimum AIC.

Sometimes there is some basis for such an ordering. If not *stepwise regression* can be used to empirically generate an ordering. There are a variety of such methods, most of which are based on the following two algorithms.

Algorithm 15.1. Forward stepwise selection:

1. Let M_0 be the null model.
2. For $k = 1, \ldots, p$, add to M_{k-1} the predictor whose addition yields the greatest reduction in SSE. This gives model M_k.
3. Select the model from $M_0, M_1, \ldots, M_p$ with the minimum AIC.

Algorithm 15.2. Backward stepwise selection:

1. Let M_p be the full model.
2. For $k = p - 1, \ldots, 0$, delete from M_{k+1} the predictor with the largest p-value. This gives model M_{k-1}.
3. Select the model from $M_0, M_1, \ldots, M_p$ with the minimum AIC.

Algorithms 15.1 or 15.1 may also be used with any other model selection score, such as BIC or R^2_{adj}.

A few technical notes are needed. Another version of the AIC, based on (15.4), is given by

$$AIC = n \log(SSE/n) + 2k,$$

which does not require an independent estimate of σ^2.

It is also worth noting the similarity between the AIC and BIC scores. Both are of the form

$$IC = -2\log(l(\hat{\theta}_{MLE})) + \lambda q$$

where $\lambda = 2$ for AIC and $\lambda = \log(n)$ for BIC. Thus, the preceding remarks concerning AIC also hold for BIC.

There is some flexibility in the definition of q. In regression applications, we get the same minimum AIC or BIC whether q is the number of predictors p or the number of parameters $p+1$ or $p+2$ (including the intercept term and the unknown variance σ^2 as appropriate).

A model selection criterion for linear regression known as *Mallow's* C_p has been proposed independently of AIC, but is equivalent to .

Finally, we note that a *finite sample* modification of the AIC has been proposed:

$$AIC_c = AIC + \frac{2k(k+1)}{n-k-1},$$

which may be used when the sample size n is relatively small. Here, it does matter whether k is the number of predictors or the number of parameters, in which case the latter should be used.

15.2 Shrinkage Methods

In the previous section, we have considered model selection methods based on subset selection. There exist a class of methods, known as *shrinkage* or *regularization* methods, which instead constrain the regression coefficients directly while fitting a full model. Suppose we define a model score for a p predictor linear regression model

$$\Lambda = SSE + \lambda \sum_{j=1}^{p} |\beta_j|^d,$$

where $\lambda \geq 0$ is a *tuning parameter* and d is a positive power. Note that the intercept β_0 is not included in the sum in the second term. We also note that such a methodology generally assumes that predictors have been standardized to zero mean and unit variance. Otherwise, summing coefficients in this manner would depend arbitrarily on the predictor's units.

This score resembles the AIC and BIC scores, in the sense that a complexity penalty is added to a standard goodness of fit measure such as SSE. However, instead of considering alternative predictor subsets, the complete model is fit for a fixed λ by minimizing Λ instead of SSE. Then λ is allowed to vary. If we set $\lambda = 0$, we have no complexity penalty, which yields the standard least squares estimates. However, as λ increases, the total magnitude of the coefficients is forced to *shrink* to 0, so that at $\lambda \to \infty$ we converge to the null model.

The usual approach is to select a relatively fine grid of λ values, $\lambda_1, \ldots, \lambda_N$. Then model M_i minimizes Λ for $\lambda = \lambda_i$. Finally, the selected model M^* of the one from the sequence $M_1, \ldots, M_N$ with the minimum value of MSE_{test}. If a simpler model is needed, the model for which the error is within 1 standard error of the cross-validated estimate of the minimum may be used.

The choice of power d is of some consequence. When $d = 2$ the method is commonly known as *ridge regression*. When $d = 1$, the method is commonly known as LASSO (least absolute shrinkage and selection operator). One of the problems with ridge regression is that all predictors remain in

the model. If a predictor does not add significantly to the model, its coefficient will be very small but not zero.

On the other hand, one of the attractive features of LASSO is that it forces some coefficients to 0, and these tend to be the ones with smaller magnitudes in a ridge regression model.

Chapter 16

Bayesian Networks

The Bayesian network (BN) is an example of a *graphical model*. First, we define a *graph*. Let V be a discrete set of *nodes* (or *vertices*). These are usually labeled, and so without loss of generality we can set $V = \{1, \ldots, n\}$, for a graph with n nodes. We also have a set of *edges*, which are pairs of nodes from V. An edge consisting of nodes i, j implies a connection between them. The edge may be defined by an unordered pair, in which case the edge is *undirected*. If the pair is ordered, then the edges defined by (i, j) and (j, i) are distinct (a graph may contain both of these edges). In this case the edge is *directed*, so that the edge (i, j) points from i to j (i is a parent, j is the child). The number of parents (children) of a node is commonly referred to as the *indegree* (*outdegree*).

A graph G, then, is simply a set of nodes V, and a set of edges from V. A *path* is a sequence of nodes $a_1, \ldots, a_m$ with edges between consecutive elements. If each edge (a_i, a_{i+1}) in the path is directed with parent a_i, then the path is directed, otherwise it is undirected.

A graph consisting of (un)directed edges is an (un)directed graph, although a graph may include both types of edges. In a directed graph, if a directed path exists from node i to node j, then i is an *ancestor* of j and j is a descendant of i. A directed path which starts and stops at the same node is a *cycle*. *Directed acyclic graphs* (DAG) are an important class of graphs, defined as any directed graph which does not contain any cycles. A family tree is an example of a DAG. Figure 16.1 shows examples of each type. Note that graph b) contains a cycle involving nodes b, c, d, g, but graph c) contains no cycles.

Formally, a BN is a random vector $\tilde{X} = (X_1, \ldots, X_n)$, and a DAG G with n nodes. The ith node is associated with the component X_i of $\tilde{X}$. The role played by G is subtle but crucial. Recall the definition of a Markov chain (Chapter 7, CSC262 lecture notes). It defines a sequence of random variables $Z_1, Z_2, Z_3, \ldots$ possessing the *memoryless property*

$$P(Z_{i+1} = a_{i+1} \mid Z_i = a_i, Z_{i-1} = a_{i-1}, \ldots, Z_1 = a_1) = P(Z_{i+1} = a_{i+1} \mid Z_i = a_i),$$

where $a_1, \ldots, a_{i+1}$ are any values that the Markov chain may assume. Intuitively, this means that the distribution of a future state Z_{i+1} given the current state Z_i and the history $Z_{i-1}, \ldots, Z_1$ depends only on the current state Z_i. In a sense, the current state separates the future and past states, formally, the future is independent of the past conditional on the present.

In a sense, a BN is a generalization of a Markov chain (and so a Markov chain is a special case of a BN). In graph c) of Figure 16.1 the direction of the edges suggests a sequential structure, with nodes a, b in one 'generation', then c, then d, then e, f, g forming subsequent generations (it

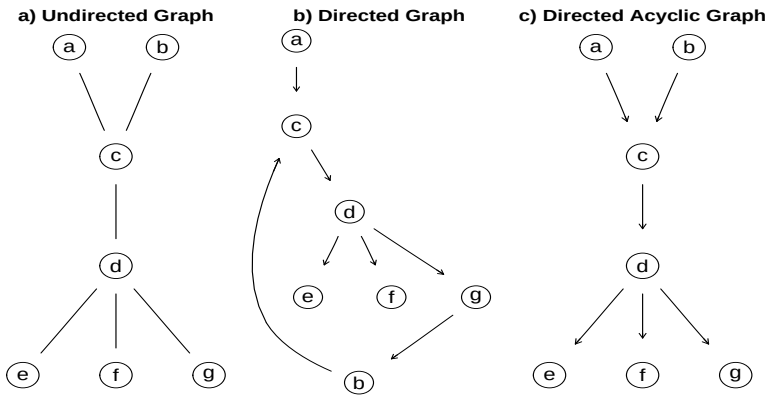

Figure 16.1: Types of graphs.

is helpful to think of a family tree in this context). There is, in fact, a version of the memorlyess property for DAGs, expressed in terms of conditional independence.

Definition 16.1. A random vector $\tilde{X} = (X_1, \ldots, X_n)$ satisfies the *local Markov property* with respect to a DAG G if each node X_i is independent of its non-descendants conditional on its parents (here X_i is used interchangeably with node i). In this case, $(\tilde{X}, G)$ is a BN.

Compare Definition 16.1 to the phrase *'the future is independent of the past conditional on the present'* used to define the memoryless property of a Markov chain. There are several ways to express the local Markov property. Consider the next definition:

Definition 16.2. Let G be a DAG, and let $\tilde{X} = (X_1, \ldots, X_n)$ be a random vector associated with the nodes of G. The *Markov blanket* of node i is the set consisting of all parents of i; all children of i; and all parents of all children of i. If each node X_i is independent of all nodes outside its Markov blanket, conditional in its Markov blanket, then $(\tilde{X}, G)$ is a BN (here X_i is used interchangeably with node i).

From a practical point of view, the advantage of the BN model is that it permits considerable simplification of the joint density $f(x_1, \ldots, x_n)$. Suppose, given DAG G, we let S_i be the set of parents of node i. Then if f satisfied either definition of a BN with respect to G (the two are equivalent) then F may be factored by

$$f(x) = \prod_{i=1}^{n} f_i(x_i \mid x_j, j \in S_i), \tag{16.1}$$

where $f_i(x_i \mid x_j, j \in S_i)$ is the density of X_i conditional on $\{X_i : i \in S_i\}$. For example, a BN based on graph c) of Figure 16.1 can be factored as

$$f = f(e \mid d)f(f \mid d)f(g \mid d)f(d \mid c)f(c \mid a, b)f(a)f(b)$$

(simplifying notation somewhat). Note that the components $f(a), f(b)$ are the marginal densities, since they have no parents. We could also say that the parents set of a and b is $\emptyset$, and write $f(a \mid \emptyset) = f(a)$ and $f(b \mid \emptyset) = f(b)$.

16.1 Fitting BNs

The factorized form of (16.1) allows standard statistical methods to be used to fit BN models. The two most common choices are Gaussian (for continuous, normally distributed observations) or multinomial (for categorical data). For Gaussian models, $f_i(x_i \mid x_j, j \in S_i)$ can be obtained by linear regression, using the parental observations as dependent variables for each node. Otherwise, $f_i(x_i \mid x_j, j \in S_i)$ is estimated by tabulating conditional probabilities. A likelihood function can therefore be defined, and AIC or BIC scores used to select a model.

16.2 Equivalence Classes

BNs are associated with causality models, but it must be remembered that it is the conditional independence constraints, and not the direction of the edges, that imply causal hypotheses (although the two are clearly related). Consider the following definition.

Definition 16.3. In a DAG, a *v-structure* is a subgraph of the form $a \rightarrow b \leftarrow c$, that is, three nodes for which two are parents of the third, and the parents are not connected by an edge. A *topology* of a DAG is the underlying undirected graph, that is, the undircted graph obtained by converting all directed edges to underected edges. Two DAGs are *equivalent*, or are members of the same *equivalence class* if they have the same v-structures and the same topology.

Two equivalent DAGs have identical conditional independence structure. In practice, this means that for a given data set, a likelihood, AIC or BIC score will be identical for BNs based on equivalent DAGs. In other words, observational data cannot be used to distinguish between equivalent DAGs.

Appendices

Appendix A

Linear Algebra

A.1 Numbers and Sets

A *set* is a collection of distinct objects of any kind. Each member of a set is referred to as an *element*, and is represented once. A set E may be *indexed*. That is, given an index set $\mathcal{T}$, each element may be assigned a unique index $t \in \mathcal{T}$, and all indices in $\mathcal{T}$ are assigned to exactly one element of E, denoted x_t. We may then write $E = \{x_t;\, t \in \mathcal{T}\}$.

The set of (finite) real numbers is denoted $\mathbb{R}$, and the set of extended real numbers is denoted $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. The restriction to nonegative real numbers is written $\mathbb{R}_+ = [0, \infty)$ and $\bar{\mathbb{R}}_+ = \mathbb{R}_+ \cup \{\infty\}$. We use standard notation for open, closed, left closed and right closed intervals (a, b), $[a, b]$, $[a, b)$, $(a, b]$. A reference to a interval I on $\bar{\mathbb{R}}$ may be any of these types.

The set of (finite) integers will be denoted $\mathbb{I}$, while the extended integers will be $\mathbb{I}_\infty = \mathbb{I} \cup \{-\infty, \infty\}$. The set of natural numbers $\mathcal{N}$ is taken to be the set of positive integers, which $\mathcal{N}_0$ is the set of nonnegative integers. A rational number is any real number expressible as a ratio of integers.

Then $\mathcal{C}$ denotes the complex numbers $z = a + bi \in \mathcal{C}$, where $i = \sqrt{-1}$, $a, b \in \mathbb{R}$ is the imaginary number. Note that i is added and multiplied as though it were a real number, in particular $i^2 = -1$. Multiplication is defined by $z_1 z_2 = (a_1 + b_1 i)(a_2 + b_2 i) = a_1 a_2 - b_1 b_2 + (a_1 b_2 + a_2 b_1)i$. The *conjugate* of $z = a + bi \in \mathcal{C}$ is written $\bar{z} = a - bi$, so that $z\bar{z} = a^2 + b^2 \in \mathbb{R}$. Together, z and $\bar{z}$, without reference to their order, form a *conjugate pair*.

The absolute value of $a \in \mathbb{R}$ is denoted $|a| = \sqrt{a^2}$, while $|z| = (z\bar{z})^{1/2} = (a^2 + b^2)^{1/2} \in \mathbb{R}$ is also known as the magnitude or modulus of $z \in \mathcal{C}$.

If $\mathcal{S}$ is a set of any type of number, $\mathcal{S}^d$, $d \in \mathcal{N}$, denotes the set of d-dimensional vectors $\tilde{s} = (s_1, \ldots, s_d)$, which are ordered collections of numbers $s_i \in \mathcal{S}$. In particular, the set of d-dimensional real vectors is written $\mathbb{R}^d$. When $0, 1 \in \mathcal{S}$, we may write the zero or one vector $\vec{0} = (0, \ldots, 0)$, $\vec{1} = (1, \ldots, 1)$, so that $c\vec{1} = (c, \ldots, c)$.

A collection of d numbers from $\mathcal{S}$ is *unordered* if no reference is made to the order (they are unlabelled). Otherwise the collection is *ordered*, that is, it is a vector. An unordered collection from $\mathcal{S}$ differs from a set in that a number $s \in \mathcal{S}$ may be represented more than once. Braces $\{\ldots\}$ enclose a set while parentheses $(\ldots)$ enclose a vector (braces will also be used to denote indexed sequences, when the context is clear).

The *cardinality* of a set E is the number of elements it contains, and is denoted $|E|$. If $|E| < \infty$ then E is a finite set. We have $|\emptyset| = 0$. If $|E| = \infty$, this statement does not suffice to characterize the cardinality of E. Two sets A, B are in a *1-1 correspondence* if a collection of pairs (a, b), $a \in A$,

$b \in B$ can be constructed such that each element of A and of B is in exactly one pair. In this case, A and B are of equal cardinality. The pairing is known as a *bijection*.

If the elements of A can be placed in a 1-1 correspondence with $\mathcal{N}$ we say A is *countable* (is *denumerable*). We also adopt the convention of referring to any subset of a countable set as countable. This means all finite sets are countable. If for countable A we have $|A| = \infty$ then A is *infinitely countable*. Note that by some conventions, the term countable is reserved for infinitely countable sets. For our purposes, it is more natural to consider the finite sets as countable.

All infinitely countable sets are of equal cardinality with $\mathcal{N}$, and so are mutually of equal cardinality. informally, a set is countable if it can be written as a list, finite or infinte. The set $\mathcal{N}^d$ is countable since, for example, $\mathcal{N}^2 = \{(1,1),(1,2),(2,1),(1,3),(2,2),(3,1),\ldots\}$. The set of rational numbers is countable, since the pairing of numerator and denominator, in any canonical representation, is a subset of $\mathcal{N}^2$.

A set A is *uncountable* (is *nondenumerable*) if $|A| = \infty$ but A is not countable. The set of real numbers, or any nonempty interval of real numbers, is uncountable.

If $A_1, \ldots, A_d$ are d sets, then $A_1 \times A_2 \times \ldots \times A_d = \times_{i=1}^{d} A_i$ is a product set, consisting of the set of all ordered selections of one element from each set $a_i \in A_i$. A vector is an element of a product set, but a product set is more general, since the sets A_i need not be equal, or even contain the same type of element. The definition may be extended to arbitrary forms of index sets.

A.2 Fields and Vector Spaces

The notion of real numbers can be generalized to that of a *field* $\mathcal{K}$, which is a set of *scalars* that is closed under the rules of addition and multiplication comparable to those available for real numbers $\mathbb{R}$. Both $\mathbb{R}$ and complex numbers $\mathcal{C}$ are fields.

A *vector space* $\mathcal{V} \subset \mathcal{K}^n$ is any set of vectors $x \in \mathcal{K}^n$ which is closed under linear and scalar composition, that is, if $x, y \in \mathcal{V}$ then $ax + by \in \mathcal{V}$ for all scalars a, b. This means the zero vector $\vec{0}$ must be in $\mathcal{V}$, and that $x \in \mathcal{V}$ implies $-x \in \mathcal{V}$.

Elements $x_1, \ldots, x_m$ of $\mathcal{K}^n$ are *linearly independent* if $\sum_{i=1}^{m} a_i x_i = 0$ implies $a_i = 0$ for all i. Equivalently, no x_i is a linear combination of the remaining vectors. The *span* of a set of vectors $\tilde{x} = (x_1, \ldots, x_n)$, denoted $span(\tilde{x})$, is the set of all linear combinations of vectors in $\tilde{x}$, which must be a vector space. Suppose the vectors in $\tilde{x}$ are not linearly independent. This means that, say, x_m is a linear combination of the remaining vectors, and so any linear combination in $span(\tilde{x})$ including x_m may be replaced with one including only the remaining vectors, so that $span(\tilde{x}) = span(x_1, \ldots, x_{m-1})$. The *dimension* of a vector space $\mathcal{V}$ is the minimum number of vectors whose span equals $\mathcal{V}$. Clearly, this equals the number in any set of linearly independent vectors which span $\mathcal{V}$. Any such set of vectors forms a *basis* for $\mathcal{V}$. Any vector space has a basis.

A.3 Equivalence Relationships

Suppose $\mathcal{X}$ is a set of objects, and $\sim$ defines a *binary relation* between two objects $x, y \in \mathcal{X}$.

Definition A.1. A binary relation $\sim$ on a set $\mathcal{X}$ is an *equivalence relation* if it satisfies the following three properties for any $x, y, z \in \mathcal{X}$:

Reflexivity $x \sim x$.

Symmetry If $x \sim y$ then $y \sim x$.

Transitivity If $x \sim y$ and $y \sim z$ then $x \sim z$.

Given an equivalence relation, an *equivalence class* is any set of the form $E_x = \{y \in \mathcal{X} \mid y \sim x\}$. If $y \in E_x$ then $E_y = E_x$. Each element $x \in \mathcal{X}$ is in exactly one equivalence class, so $\sim$ induces a partition of $\mathcal{X}$ into equivalence classes.

In Euclidean space, 'is parallel to' is an equivalence relation, while 'is perpendicular to' is not.

For finite sets, cardinality is a property of a specific set, while for infinite sets, cardinality must be understood as an equivalence relation.

A.4 Matrices

Let $M_{m,n}(\mathcal{K})$ be the set of $m \times n$ matrices A, for which $A_{i,j} \in \mathcal{K}$ (or, when required for clarity, $[A]_{i,j} \in \mathcal{K}$) is the element of the ith row and jth column. When the field need not be given, we will write $M_{m,n} = M_{m,n}(\mathcal{K})$. We will generally be interested in $M_{m,n}(\mathcal{C})$, noting that the real matrices $M_{m,n}(\mathbb{R}) \subset M_{m,n}(\mathcal{C})$ can be considered a special case of complex matrices, so that any resulting theory holds for both types. This is important to note, since even when interest is confined to real valued matrices, complex numbers enter the analysis in a natural way, so it is ultimately necessary to consider complex vectors and matrices. Definitions associated with real matrices (transpose, symmetric, and so on) have analgous definitions for complex matrices, which reduce to the more familiar definitions when the matrix is real.

The *square matrices* are denoted as $M_m = M_{m,m}$. Elements of $M_{m,1}$ are *column vectors* and elements of $M_{1,m}$ are *row vectors*. A matrix in $M_{m,n}$ is equivalently an ordered set of m row vectors or n column vectors. The transpose $A^T \in M_{n,m}$ of a matrix $A \in M_{m,n}$ has elements $A'_{j,i} = A_{i,j}$. For $A \in M_{n,k}$, $B \in M_{k,m}$ we always understand matrix multiplication to mean that $C = AB$ possesses elements $C_{i,j} = \sum_{k'=1}^{k} A_{i,k'} B_{k',j}$, so that matrix multiplication is generally not commutative (a binary operation $\circ$ is *commutative* if $a \circ b = b \circ a$ for all pairs (a, b) for which the operation is defined). Then $(A^T)^T = A$ and $(AB)^T = B^T A^T$ where the product is permitted.

In the context of matrix algrebra, a vector $x \in \mathcal{K}^n$ is usually assumed to be a column vector in $M_{n,1}$. Therefore, if $A \in M_{m,n}$ then the expression Ax is understood to be evaluated by matrix multiplication. Similarly, if $x \in \mathcal{K}^m$ we may use the expression $x^T A$, understanding that $x \in M_{m,1}$.

When $A \in M_{m,n}(\mathcal{C})$, the *conjugate matrix* is written $\bar{A}$, and is the component-wise conjugate of A. The identity $\bar{A}\bar{B} = \overline{AB}$ holds. The *conjugate transpose* (or *Hermitian adjoint*) of A is $A^* = \bar{A}^T$. As with the transpose operation, $(A^*)^* = A$ and $(AB)^* = B^* A^*$ where the product is permitted. This generally holds for arbitrary products, that is $(ABC)^* = (BC)^* A^* = C^* B^* A^*$, and so on. For $A \in M_{m,n}(\mathbb{R})$, we have $A = \bar{A}$ and $A^* = A^T$, so the conjugate transpose may be used in place of the transpose operation when matrices are real valued. We always may write $(A + B)^* = A^* + B^*$ and $(A + B)^T = A^T + B^T$ where dimensions permit.

A matrix $A \in M_n(\mathcal{C})$ is *diagonal* if the only nonzero elements are on the diagonal, and can therefore be referred to by the diagonal elements $diag(a_1, \ldots, a_n) = diag(A_{1,1}, \ldots, A_{n,n})$. A diagonal matrix is *positive diagonal* or *nonnegative diagonal* if all diagonal elements are positive or nonegative.

The identity matrix $I \in M_m$ is the matrix uniquely possessing the property that $A = IA = AI$ for all $A \in M_m$. For $M_m(\mathcal{C})$, I is diagonal, with diagonal entries equal to 1. For any matrix $A \in M_m$

there exists at most one matrix $A^{-1} \in M_m$ for which $AA^{-1} = I$, referred to as the *inverse* of A. An inverse need not exist (for example, if the elements of A are constant).

The *inner product* (or *scalar product*) of two vectors $x, y \in \mathcal{C}^n$ is defined as $\langle x, y\rangle = y^*x$. For any $x \in \mathcal{C}^n$ we have $\langle x, x\rangle = \sum_i \bar{x}_i x_i = \sum_i |x_i|^2$, so that $\langle x, x\rangle$ is a nonnegative real number, and $\langle x, x\rangle = 0$ if and only if $x = \vec{0}$. The magnitude, or *norm*, of a vector may be taken as $\|x\| = (\langle x, x\rangle)^{1/2}$.

Two vectors $x, y \in \mathcal{C}^n$ are *orthogonal* if $\langle x, y\rangle = 0$. A set of vectors $x_1, \ldots, x_m$ is orthogonal if $\langle x_i, x_j\rangle = 0$ when $i \neq j$. A set of m orthogonal vectors are linearly independent, and so form the basis for an m dimensional vector space. If in addition $\|x_i\| = 1$ for all i, the vectors are *orthonormal*.

A matrix $Q \in M_n(\mathcal{C})$ is *unitary* if $Q^*Q = QQ^* = I$. Equivalently, Q is unitary if and only (i) it's column vectors are orthonormal; (ii) it's row vectors are orthonormal; (iii) it possesses inverse $Q^{-1} = Q^*$. The more familiar term *orthogonal matrix* is usually reserved for a real valued unitary matrix (otherwise the definition need not be changed).

A unitary matrix preserves magnitude, since $\langle Qx, Qx\rangle = (Qx)^*(Qx) = x^*Q^*Qx = x^*Ix = x^*x = \|x\|^2$.

A matrix $Q \in M_n(\mathcal{C})$ is a *permutation* matrix if each row and column contains exactly one 1 entry, with all other elements equal to 0. Then $y = Qx$ is a permutation of the elements of $x \in \mathcal{C}^n$. A permutation matrix is always orthogonal.

Suppose $A \in M_{m,n}$ and let $\alpha \subset \{1, \ldots, m\}$, $\beta \subset \{1, \ldots, n\}$ be any two nonempty subsets of indices. Then $A[\alpha, \beta] \in M_{|\alpha|,|\beta|}$ is the *submatrix* of A obtained by deleting all elements except for $A_{i,j}$, $i \in \alpha$, $j \in \beta$. If $A \in M_n$, and $\alpha = \beta$, then $A[\alpha, \alpha]$ is a *principal submatrix*.

The determinant associates a scalar with $A \in M_m(\mathcal{C})$ through the recursive formula

$$\det(A) = \sum_{i=1}(-1)^{i+j} A_{i,j} \det(A^{i,j}) = \sum_{j=1}(-1)^{i+j} A_{i,j} \det(A^{i,j})$$

where $A^{i,j} \in M_{m-1}(\mathcal{C})$ is the matrix obtained by deleting the ith row and jth column of A. Note that in the respective expressions any j or i may be chosen, yielding the same number, although the choice may have implications for computational efficiency. As is well known, for $A \in M_1(\mathcal{C})$ we have $\det(A) = A_{1,1}$ and for $A \in M_2$ we have $\det(A) = A_{1,1}A_{2,2} - A_{1,2}A_{2,1}$. In general, $\det(A^T) = \det(A)$, $\det(A^*) = \overline{\det(A)}$, $\det(AB) = \det(A)\det(B)$, $\det(I) = 1$ which implies $\det(A^{-1}) = \det(A)^{-1}$ when the inverse exists.

A large class of algorithms is associated with the problem of determining a solution $x \in \mathcal{K}^m$ to the *linear systems of equations* $Ax = b$ for some fixed $A \in M_m$ and $b \in \mathcal{K}^m$.

Theorem 4. The following statements are equivalent for $A \in M_m(\mathcal{C})$, and a matrix satisfying any one is referred to as *nonsingular*, any other matrix in $M_m(\mathcal{C})$ *singular*:

(i) The columns vectors of A are linearly independent.

(ii) The row vectors of A are linearly independent.

(iii) $\det(A) \neq 0$.

(*iv*) $Ax = b$ possesses a unique solution for any $b \in \mathcal{K}^m$.

(*v*) $x = \vec{0}$ is the only solution of $Ax = \vec{0}$.

Matrices $A, B \in M_n$ are *similar*, if there exists a nonsingular matrix S for which $B = S^{-1}AS$. Simlarity is an equivalence relation (Section A.3). A matrix is *diagonalizable* if it is similar to a diagonal matrix. Diagonalization offers a number of advantages. We always have $B^k = S^{-1}A^kS$, so that if A is diagonal, this expression is particularly easy to evaluate. More generally, diagonalization can make apparent the behavior of a matrix interpreted as a transformation. Suppose in the diagonalization $B = S^{-1}AS$ we know that S is orthogonal, and that A is real. Then the action of B on a vector is decomposed into S (a change in coordinates), A (elementwise scalar multiplication) and S^{-1} (the inverse change in coordinates).

A.5 Eigenvalues and Spectral Decomposition

For $A \in M_n(\mathcal{C})$, $x \in \mathcal{C}^n$, and $\lambda \in \mathcal{C}$ we may define the *eigenvalue equation*

$$Ax = \lambda x, \tag{A.1}$$

and if the pair (λ, x) is a solution to this equation for which $x \neq \vec{0}$, then λ is an *eigenvalue* of A and x is an associated *eigenvector* of λ. Any such solution (λ, x) may be called an *eigenpair*. Clearly, if x is an eigenvector, so is any nonzero scalar multiple. Let R_λ be the set of all eigenvectors x associated with λ, If $x, y \in R_\lambda$ then $ax+by \in R_\lambda$, so that R_λ is a vector space. The dimension of R_λ is known as the *geometric multiplicity* of λ. We may refer to R_λ as an *eigenspace* (or *eigenmanifold*). In general, the *spectral properties* of a matrix are those pertaining to the set of eigenvalues and eigenvectors.

If $A \in M_n(\mathbb{R})$, and λ is an eigenvalue, then so is $\bar{\lambda}$, with associated eigenvectors $R_{\bar{\lambda}} = \bar{R}_\lambda$. Thus, in this case eigenvalues and eigenvectors occur in conjugate pairs. Simlarly, if λ is real there exists a real associated eigenvector.

The eigenvalue equation may be written $(A - \lambda I)x = 0$. However, by Theorem 4 this has a nonzero solution if and only if $A - \lambda I$ is singular, which occurs if and only if $p_A(\lambda) = \det(A - \lambda I) = 0$. By construction of a determinant, $p_A(\lambda)$ is an order n polynomial in λ, known as the *characteristic polynomial* of A. The set of all eigenvalues of A is equivalent to the set of solutions to the *characteristic equation* $p_A(\lambda) = 0$ (including complex roots). The multiplicity of an eigenvalue λ as a root of $p_A(\lambda)$ is referred to as it's *algebraic multiplicity*. A *simple eigenvalue* has algebraic multiplicity 1. The geometric multiplicity of an eigenvalue can be less, but never more, than the algebraic multiplicity. A matrix with equal algebraic and geometric multiplicities for each eigenvalue is a *nondefective matrix*, and is otherwise a *defective matrix*.

We therefore denote the set of all eigenvalues as $\sigma(A)$. An important fact is that $\sigma(A^k)$ consists exactly of the eigenvalues $\sigma(A)$ raised to the kth power, since if (λ, x) solves $Ax = \lambda x$, then $A^2x = A\lambda x = \lambda Ax = \lambda^2 x$, and so on. A quantity of particular importance is the *spectral radius* $\rho(A) = \max\{|\lambda| \mid \lambda \in \sigma(A)\}$. There is sometimes interest in ordering the eigenvalues by magnitude. If there exists an eigenvalue $\lambda_1 = \rho(A)$, this is sometimes referred to as the *principal eigenvalue*, and any associated eigenvector is a *principle eigenvector*.

Suppose we may construct n eigenvalues $\lambda_1, \ldots, \lambda_n$, with associated eigenvectors $\nu_1, \ldots, \nu_n$. Then let $\Lambda \in M_n$ be the diagonal matrix with ith diagonal element λ_i, and let $V \in M_n$ be the

matrix with ith column vector ν_i. By virtue of (A.1) we can write

$$AV = V\Lambda. \tag{A.2}$$

If V is invertable (equivalently, there exist n linearly independent eigenvectors, by Theorem 4), then

$$A = V\Lambda V^{-1}, \tag{A.3}$$

so that A is diagonalizable. Alternatively, if A is diagonalizable, then (A.2) can be obtained from (A.3) and, since V is invertable, there must be n independent eigenvectors. The following theorem expresses the essential relationship between diagonalization and spectral properties.

Theorem 5. For square matrix $A \in M_n(\mathcal{C})$:

(i) Any set of $k \leq n$ eigenvectors $\nu_1, \ldots, \nu_k$ associated with distinct eigenvalues $\lambda_1, \ldots, \lambda_k$ are linearly independent,

(ii) A is diagonalizable if and only if there exist n linearly independent eigenvectors,

(iii) If A has n distinct eigenvalues, it is diagonalizable (this follows from (i) and (ii)),

(iv) A is diagonalizable if and only if it is nondefective.

A.5.1 Right and Left Eigenvectors

The eigenvectors defined by (A.1) may be referred to as *right eigenvectors*, while *left eigenvectors* are nonzero solutions to

$$x^*A = \lambda x^*, \tag{A.4}$$

(note that some conventions do not explicitly refer to complex conjugates x^* in (A.4)). This similarly leads to the equation $x^*(A - \lambda I) = 0$, which by an argument identical to that used for right eigenvectors, has nonzero solutions if and only if $p_A(\lambda) = 0$, giving the same set of eigenvalues as those defined by (A.1). There is therefore no need to distinguish between 'right' and 'left' eigenvalues. Then, fixing eigenvalue λ we may refer to the *left eigenspace* L_λ as the set of solution x to (A.4) (in which case, R_λ now becomes the *right eigenspace* of λ).

The essential relationship between the eigenspaces is summarized in the following theorem:

Theorem 6. Suppose $A \in M_n(\mathcal{C})$.

(i) For any $\lambda \in \sigma(A)$ L_λ and R_λ have the same dimension.

(ii) For any distinct eigenvalues $\lambda_1, \ldots, \lambda_m$ from $\sigma(A)$, any selection of vectors $x_i \in R_{\lambda_i}$ for $i = 1, \ldots, m$ are linearly independent. The same holds for selections from distinct L_λ.

(iii) Right and left eigenvectors associated with distinct eigenvalues are orthogonal.

Proof. Proofs may be found in, for example, Chapter 1 of *Matrix Analysis*, Horn and Johnson, 1985. □

Next, if V is invertible, multiply both sides of (A.3) by V^{-1} yielding

$$V^{-1}A = \Lambda V^{-1}.$$

Just as the column vectors of V are right eigenvectors, we can set $U^* = V^{-1}$, in which case the ith column vector υ_i of U is a solution x to the left eigenvector equation (A.4) corresponding to eigenvalue λ_i (the ith element on the diagonal of Λ). This gives the diagonalization

$$A = V\Lambda U^*.$$

Since $U^*V = I$, indefinite multiplication of A yields the *spectral decomposition*:

$$A^m = V\Lambda^m U^* = \sum_{i=1}^{n} \lambda_i^m \nu_i \upsilon_i^*. \tag{A.5}$$

The apparent recipe for a spectral decomposition is to first determine the roots of the characteristic polynomial, and then to solve each resulting eigenvalue equation (A.1) after substituting an eigenvalue. This seemingly straightforward procedure proves to be of little practical use in all but the simplest cases, and spectral decompositions are often difficult to construct using any method. However, a complete spectral decomposition need not be the objective. First, it may not even exist for many otherwise interesting models. Second, there are many important problems related to A that can be solved using spectral theory, but without the need for a complete spectral decomposition. For example:

(*i*) Determining bounds $\|Ax\| \leq a\,\|x\|$ or $\|Ax\| \geq b\,\|x\|$,

(*ii*) Determining the convergence rate of the limit $\lim_{k\to\infty} A^k = A^\infty$,

(*iii*) Verifying the existence of a scalar λ and vector ν for which $A\nu = \lambda\nu$, and guaranteeing that (for example) λ and ν are both real and positive.

Basic spectral theory relies on the identification of special matrix forms which impose specific properties on a the spectrum. We next discuss two cases.

A.6 Symmetric, Hermitian and Positive Definite Matrices

A matrix $A \in M_n(\mathcal{C})$ is *Hermitian* if $A = A^*$. A Hermitian real valued matrix is *symmetric*, that is, $A = A^T$. The spectral properties of Hermitian matrices are quite definitive (see, for example, Chapter 4, *Matrix Analysis*, Horn and Johnson, 1985).

Theorem 7. A matrix $A \in M_n(\mathcal{C})$ is Hermitian if and only if there exists a unitary matrix U and real diagonal matrix Λ for which $A = U\Lambda U^*$.

A matrix $A \in M_n(\mathbb{R})$ is symmetric if and only if there exists a real orthogonal Q and real diagonal matrix Λ for which $A = Q\Lambda Q^T$.

Clearly, the matrices Λ and U may be identified with the eigenvalues and eigenvectors of A, with n eignevalue equation solutions given by the respect columns of $AU = U\Lambda U^*U = U\Lambda$. An important implication of this is that all eigenvalues of a Hermitian matrix are real, and eigenvectors may be selected to be orthonormal.

If we interpet $x \in \mathcal{C}^n$ as a column vector $x \in M_{n,1}$ we have *quadratic form* x^*Ax, which is interpretable either as a 1×1 complex matrix, or as a scalar in $\mathcal{C}$, as is convenient.

If A is Hermitian, then $(x^*Ax)^* = x^*A^*x = x^*Ax$. This means if $z = x^*Ax \in \mathcal{C}$, then $z = \bar{z}$, equivalently $x^*Ax \in \mathbb{R}$. A Hermitian matrix A is *positive definite* if and only if $x^*Ax > 0$ for all $x \neq \vec{0}$. If instead $x^*Ax \geq 0$ then A is *positive semidefinite*. A nonsymmetric matrix satisfying $x^TAx > 0$ can be replaced by $A' = (A + A^T)/2$, which is symmetric, and also satisfies $x^TA'x > 0$.

Theorem 8. If $A \in M_n(\mathcal{C})$ is Hermitian then x^*Ax is real. If, in addition, A is positive definite then all of its eigenvalues are positive. If it is positive semidefinite then all of its eigenvalues are nonnegative.

If A is positive semidefinite, and we let λ_{min} and λ_{max} be the smallest and largest eigenvalies in $\sigma(A)$ (all of which are nonnegative real numbers) then it can be shown that

$$\lambda_{min} = \min_{\|x\|=1} x^*Ax \text{ and } \lambda_{max} = \max_{\|x\|=1} x^*Ax.$$

If A is positive definite then $\lambda_{min} > 0$. In addition, since the eigenvalues of A^2 are the squares of the eigenvalues of A, and since for a Hermitian matrix $A^* = A$, we may also conclude

$$\lambda_{min} = \min_{\|x\|=1} \|Ax\| \text{ and } \lambda_{max} = \max_{\|x\|=1} \|Ax\|,$$

for any positive semidefinite matrix A.

Any diagonalizable matrix A possesses a kth root, $A^{1/k}$, meaning $A = \left(A^{1/k}\right)^k$. Given diagonalization $A = Q^{-1}\Lambda Q$, this is easily seen to be $A^{1/k} = Q^{-1}\Lambda^{1/k}Q$, where $[\Lambda^{1/k}]_{i,j} = (\Lambda_{i,j})^{1/k}$. If A is a real symmetric positive definite matrix then $A^{1/2}$ is real, symmetric and nonsingular.

Appendix B

Multivariate Distributions

We will need to characterize the distribution of random vectors $\tilde{X} = (X_1, \ldots, X_n)$, described in terms of densities, PMFs and CDFs. A random vector can have components of different types (say, X_1 is discrete and X_2 is continuous). However, the usual approach is to assume that all components are of the same type. Once the theory is understood under this restriction, extension to the more general case will be quite natural.

A *discrete* random vector $\tilde{X} = (X_1, \ldots, X_n)$ possesses a PMF which assigns a probability $p_{\tilde{X}}(x_1, \ldots, x_n) \in [0, 1]$ to each element $\tilde{x} = (x_1, \ldots, x_n)$ of a support set $\mathcal{S}$ such that

$$\sum_{\tilde{x} \in \mathcal{S}} p_{\tilde{X}}(\tilde{x}) = 1. \tag{B.1}$$

Then the probability of $E \subset \mathbb{R}^n$ is

$$P(E) = \sum_{\tilde{x} \in E \cap \mathcal{S}} p_{\tilde{X}}(\tilde{x}). \tag{B.2}$$

A *continuous* random vector $\tilde{X} = (X_1, \ldots, X_n)$ possess a density function $f_{\tilde{X}}(x_1, \ldots, x_n) \leq 0$ which satisfies the condition

$$\int_{\tilde{x} \in \mathbb{R}^n} f_{\tilde{X}}(\tilde{x}) d\tilde{x} = 1. \tag{B.3}$$

Then the probability of $E \subset \mathbb{R}^n$ is

$$P(E) = \int_{\tilde{x} \in E} f_{\tilde{X}}(\tilde{x}) d\tilde{x}. \tag{B.4}$$

The support of $\tilde{X}$ consists of all points $\tilde{x}$ for which $f_{\tilde{X}}(\tilde{x}) > 0$.

Suppose the components of $\tilde{X}$ are independent, and X_i has density f_i. Then the joint density of $\tilde{X}$ is

$$f_{\tilde{X}}(x_1, \ldots, x_n) = \prod_{i=1}^{n} f_i(x_i). \tag{B.5}$$

B.1 Matrix Algebra and Multivariate Distributions

Suppose we have random vector $\tilde{X} = (X_1, \ldots, X_m)$. The *mean vector* can be written $\mu_{\tilde{X}} = E[\tilde{X}] = (E[X_1], \ldots, E[X_m])$ in the appropriate context. In matrix algebra $\mu_{\tilde{X}}$ is usually interpreted as a column vector.

The $m \times m$ *variance matrix* (also referred to as the *covariance matrix*) of $\tilde{X}$ is defined elementwise as

$$\left[\Sigma_{\tilde{X}}\right]_{i,j} = cov\left[X_i, X_j\right],$$

where we denote covariance $cov\left[X, Y\right] = E[(X - E[X])(Y - E[Y])]$ and consequently $var\left[X\right] = cov\left[X, X\right]$. Two RVs may be refered to as *linearly independent* if their covariance is zero, although this does not by itself imply independence under the formal definition.

When the context permits we may write $var\left[\tilde{X}\right] = \Sigma_{\tilde{X}}$. Since $cov\left[X, Y\right] = cov\left[Y, X\right]$, $\Sigma_{\tilde{X}}$ is always symmetric. For any linear combination $Y = a_1 X_1 + \ldots + a_m X_m$ based on constant coefficients a_i it may be shown that

$$var\left[Y\right] = \tilde{a}^T \Sigma_{\tilde{X}} \tilde{a}, \tag{B.6}$$

where $\tilde{a} = (a_1, \ldots, a_m)$ is taken to be a column vector. Since a variance is always nonnegative this must mean $\Sigma_{\tilde{X}}$ is positive semidefinite, and is positive definite unless a subset of the elements of $\tilde{X}$ are linearly dependent with probability 1.

Next, suppose b is a $k \times 1$ constant column vector, A is a $k \times m$ constant matrix, and $\tilde{X}$ is a $m \times 1$ random vector. Then

$$\tilde{Y} = b + A\tilde{X}$$

is a linear tranformation yielding a $k \times 1$ random vector, consisting of k linear combinations of $\tilde{X}$. The mean and variance matrices of $\tilde{X}$ and $\tilde{Y}$ are always related by

$$E[\tilde{Y}] = b + AE[\tilde{X}] \text{ and } var\left[\tilde{Y}\right] = A\left(var\left[\tilde{X}\right]\right)A^T.$$

Suppose $var\left[\tilde{X}\right]$ is positive definite. Then there exists an invertible symmetric square root matrix $var\left[\tilde{X}\right]^{1/2}$ (Section A.6). If $\tilde{Y} = var\left[\tilde{X}\right]^{-1/2}\tilde{X}$ then

$$\begin{aligned} var\left[\tilde{Y}\right] &= var\left[\tilde{X}\right]^{-1/2} var\left[\tilde{X}\right] var\left[\tilde{X}\right]^{-1/2} \\ &= var\left[\tilde{X}\right]^{-1/2} var\left[\tilde{X}\right]^{1/2} var\left[\tilde{X}\right]^{1/2} var\left[\tilde{X}\right]^{-1/2} \\ &= I. \end{aligned}$$

Thus, any random vector with a positive definite variance matrix $var\left[\tilde{X}\right]$ possesses a linear transformation yielding linearly independent coordinates of unit variance.

B.2 Multivariate Normal Distribution

Suppose $\tilde{\mu}$ is a $m \times 1$ column vector and Σ is a positive defnite $m \times m$ matrix. The *multivariate normal density* function is defined as

$$\begin{aligned} f(x \mid \tilde{\mu}, \Sigma) &= (2\pi)^{-m/2} \det(\Sigma)^{-1/2} \exp(-Q/2), \;\; x \in \mathbb{R}^m, \text{ where} \\ Q &= (x - \tilde{\mu})^T \Sigma^{-1} (x - \tilde{\mu}). \end{aligned} \tag{B.7}$$

Then $\tilde{X} = (X_1, \ldots, X_m)$ is a *multivariate normal random vector* if it possesses this density, in which case it may be shown that $E[\tilde{X}] = \tilde{\mu}$, $var\left[\tilde{X}\right] = \Sigma$. In addition, the marginal distributions are $X_i \sim N(\tilde{\mu}_i, \Sigma_{i,i})$. The $m = 2$ case is often referred to as the *bivariate normal* distribution.

It is important to note that a random vector with marginal normal densities is not necessarily multivariate normal. For example, if $X \sim N(0, 1)$ and $Y = SX$ where S is an independent random sign, then $Y \sim N(0, 1)$, $cov\,[X, Y] = 0$, but (X, Y) does not possess a multivariate normal density.

The definition of a multivariate normal random vector can be generalized to include any random vector of the form $\tilde{X} = \tilde{\mu} + A\tilde{Z}$, where $\tilde{\mu}$ is an $k \times 1$ column vector, A is an $k \times m$ matrix, and $\tilde{Z}$ is a $m \times 1$ column vector of independent unit normal random variables. In this case $var\left[\tilde{X}\right]$ need not be positive definite, so (B.7) cannot be used directly.